電腦時代

COMPUTING AGE

16

工商業●教育●家庭娛樂●私人應用電腦雜誌 港幣十元

Quality *Dysan* Diskettes in Distinctive Colors



Authorized Distributor:

Supply Consultants Ltd.
Room 1802, Valley Centre 18/F.
80–82 Morrison Hill Road, Hong Kong.
Tel: 5-8917697 Telex: 61627 SUPAV HX

DEALERS:
EXCELLENT ELECTRONIC CO. LTD TEL. 3-313878
MICROGRAM COMPUTER COMPANY TEL: 3-7282113

Super 5

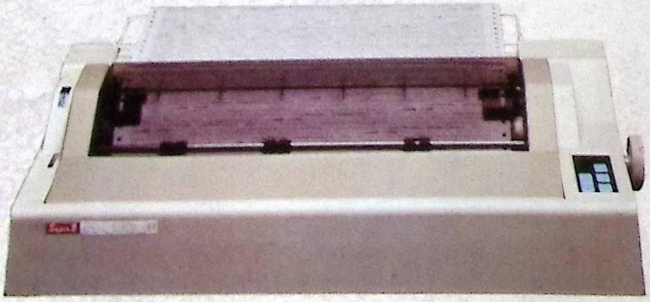
DOT MATRIX PRINTER
EN-P1091/EN-P1090
点阵式打印机
NEW VERSION
EP – 1201



EN-P1091
- Print Mode Selector Switch
- Near-letter quality printing
- Print Speed 120 cps.

EN-P1090
- High resolution, dot-addressable graphics
- Print speed 80 cps.

100% IBM* PC COMPATIBLE



SUPER 5 MBE-420 15" PRINTER • Print speed 180 cps.



SUPER 5 EP-1807 15" 7 COLOR PRINTER
- Inked platens printer

NEW

SUPER 5 CP80/CPA80 9 pin printer

SUPER 5 MONITOR FOR IBM PC 资料显示器

SUPER 5 POWER SUPPLY 变压器
for IBM* pc Max. output Power 133.9W

SUPER 5 5.25" Floppy Diskette – EE50(SSDD)/EN-320K (DSDD)

SUPER T-40e/C-40 DISK DRIVE 软盘磁碟驱动器
Plus FDD for IBM* PC

CHINON 502 / EPSON SP-521 / TEAC FD-55B / MATSUSHITA JA551-2

* IBM PC is a registered trademark of International Business Machine Corp.

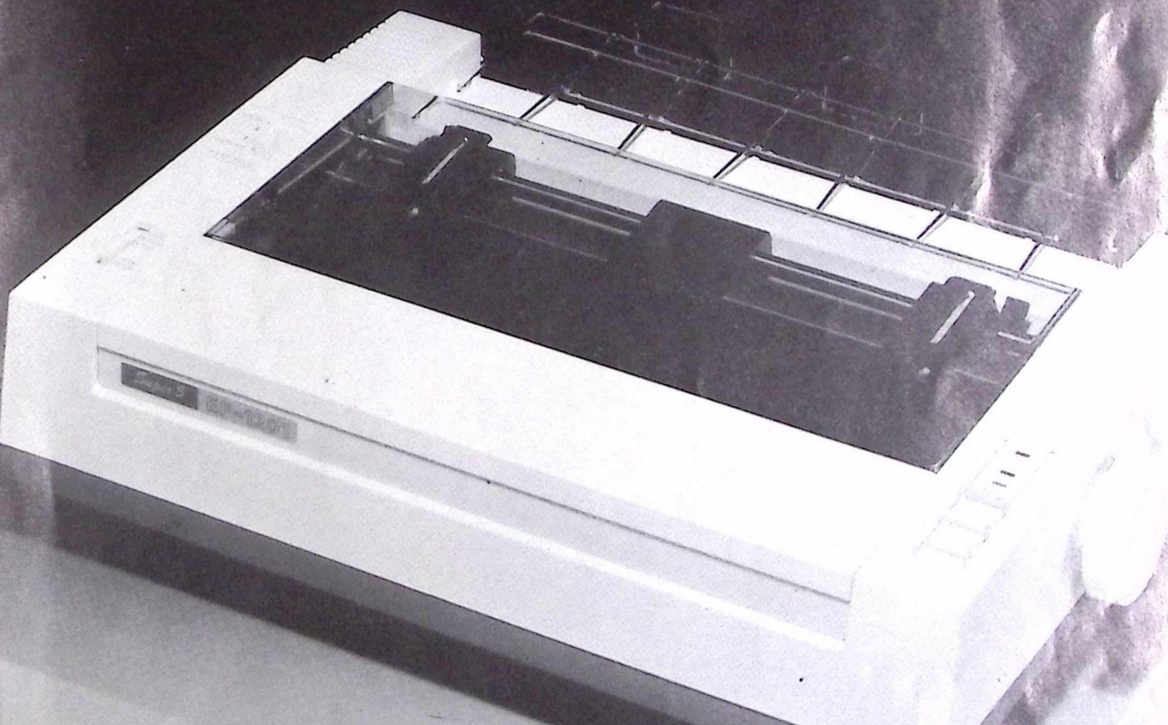
Super 5
EI-EN ELECTRONICS, LTD.
22-23-213 Ebisu-I-Chome Shibuya-Ku, Tokyo, Japan.
Telex J23325 EIEN ENT FAX (03) 446-6530.

A & N ENTERPRISE CO., LTD.
20/F., Western Centre, 40-50 Des

Super 5 EP-1201

(NEW VERSION OF EN-P1091)

Impact Dot Matrix Printer



Super 5

A & N ENTERPRISE CO., LTD.

20/F WESTERN CENTRE., 40-50 DES VOEUX RD, WEST, H.K.
TEL: 5-461179 5-488463 TELEX: 65218 MCLWW HX
FAX: 5—497149


TAXAN RGB「幻彩系列」
彩色電腦螢幕
色彩艷麗完美無瑕
● 能顯示任何顏色組合
率先採用「線性影像放大系統」
● 絕不干擾磁碟及其他電器
設計符合多項美國嚴格標準
● 適用於各款電腦
備有多種附件，適用於下列電腦：
RGB（紅綠藍）彩色介面
APPLE II [1] APPLE IIe [1]（連80字顯示）
介面接駁線
APPLE III [1] IBM PC [2] BBC [3] NEC PC [4]
FUJITSU FM-7 [5] COMMODORE [6]
歡迎訂造特別介面接駁線
享有代理
才是
RGB vision-III
TAXAN
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

精益求精
擺脫傳統設計
TAXAN® 再放新異彩

今日的電腦設備不單祗在功能上加速發展，TAXAN® 更兼顧外形及實用性設計，最近推出 K X 系列綠色電腦螢幕更具下列優點：

* 不反光方形平面深底色螢幕設計，解像力特高，視頻高達20 MHz，畫面更清晰。
* 線條優美，備有多種外置調校功能，適合不同環境使用。
* 備有多種型號可供選擇。
  K X 1201 －（P 31）深綠底色顯像管，適合一般電腦。
  K X 1212 －（P 39）深灰底色長殘影顯像管，適合IBM PC*及同類型產品。
  其他顏色顯像管，歡迎訂購。
  請駕臨各大分銷商參觀比較。

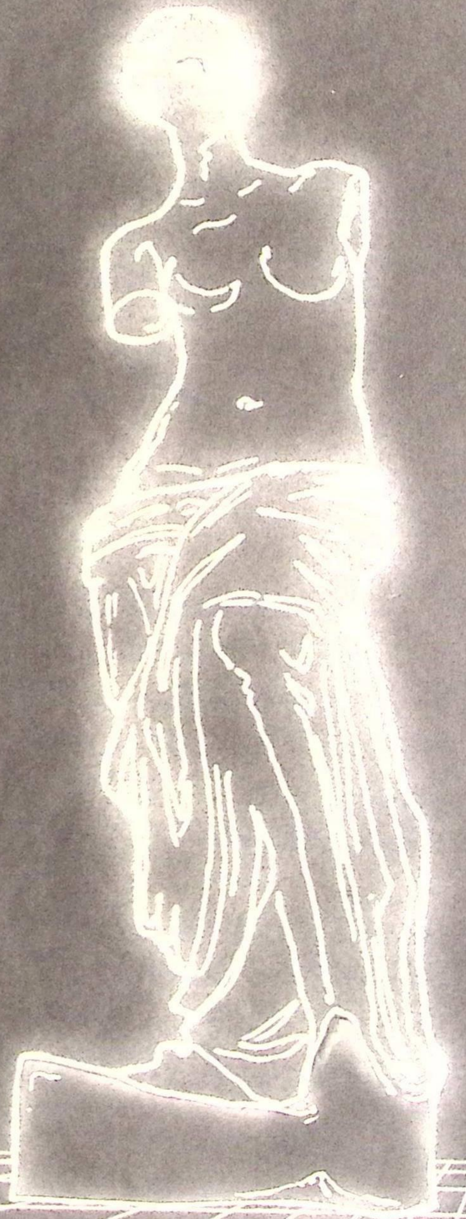
*IBM PC is a trademark of International Business Machine Corp.

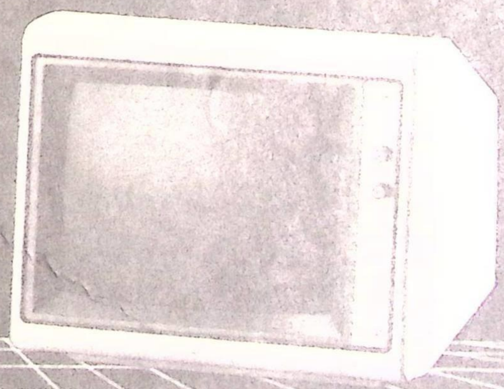
享有代理商保証，才是專業性保用。謹防水貨及二手貨，請索取全新專用保用証，証上均註明該貨品型號。如有疑問，請立即向本公司查詢。

總代理：

VINCENT COMPUTER CENTRE LTD
雲生電腦有限公司

九龍尖沙咀柯士甸路22-26號好兆年行6字樓 電話：3-7211268 0 電報：HX 36310 VCC HK



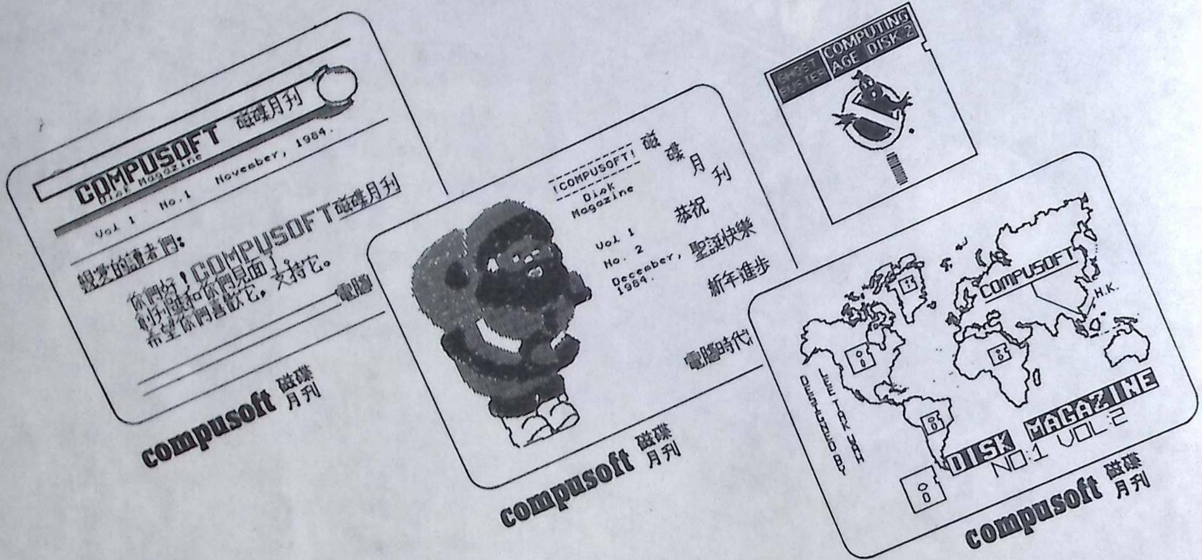


請即訂閱

一本眞眞正正的電腦雜誌

蘋果電腦用家絕不容錯過的雜誌

COMPUSOFT 磁碟月刊經已出版



訂閱磁碟月刊　小小的代價！　多多的得益！

電腦時代出版社爲蘋果電腦用家的劃時代貢獻，只要你是蘋果電腦用家，就必定要擁有COMPUSOFT 磁碟月刊。你可從磁碟月刊裡找到其他電腦雜誌絕不能做到的內容。COMPUSOFT 磁碟月刊內容取材廣泛，包括教育軟件，實用工具程式，電腦自學文章，最新軟件出版消息，螢幕顯示表達技巧，精彩娛樂軟件　，ADVENTURE 破解方法等等，適合電腦發燒友，未來電腦從業員，學校教職員，學生，關心子女學業的家長及有志投身電腦軟件工業的人士。

COMPUSOFT 磁碟月刊是一本眞眞正正的電腦雜誌，每期內內容必定的能滿足你的需求，而磁碟月刊本身用來表達內容的技巧，更是我們一羣專業程序設計師精心泡製的成果，有極大的硏究價值。今日就請你立即訂閱。訂閱COMPUSOFT 磁碟月刊，只要花小小的代價，就獲得多多的好處。

COMPUSOFT磁碟月刊每月出版一次，每次出版兩張雙面灌錄的磁碟，以DOS 3.3 FORMAT，適用於任何蘋果電腦或相容的機種。

COMPUSOFT磁碟月刊只設長期訂閱，不公開發售，請立即填妥訂閱表格寄來。

訂閱辦法

訂閱COMPUSOFT 磁碟月刊辦法簡單，訂戶每期收到磁碟後，自行COPY一份，然後寄回磁碟給我們，我們便會在下期出版之時繼續寄出。當然，訂戶亦可於每月出版之時前來電腦時代出版社讀者服務部攞取每期月刊。

讀者如對訂閱辦法有任何疑問，可致電3-312007查詢。

海外讀者只接受連同磁碟的訂閱，請附加空郵費用20%。澳門讀者亦只接受連同磁碟的訂閱，請附加多10%郵費。

只適用於本港訂戶

本人欲訂閱 COMPUSOFT 磁碟月刊

☐ 12期　HK $180（退回磁碟）
☐ 12期　HK $540（連同磁碟）*
☐ 6期　HK $100（退回磁碟）
☐ 6期　HK $280（連同磁碟）*
☐ 1期試閱　HK $ 50（連同磁碟）*

由第___期開始

支票收款人請填寫 "Computing Age Publisher"
請寄九龍中央郵箱71193號
信封面請註明「訂閱」磁碟月刊

*磁碟歸訂戶擁有

姓名：
地址：
電話：
附上劃線支票 No.
（請用英文填寫，方便電腦處理）

NO.
Official Use

第2期

陽曆陰曆日子對算表資料輸入法

創刊號中登載的〔西曆査農曆日子〕只能査閱1924年至1986年之間的日子，以後那些年份便沒有效。事實本程式並沒有年份限制，今期我們教你如何輸入1987年以後年份的資料來查閱。

磁碟雜誌中文字顯示原理剖析（一）

磁碟雜誌的內容說明主要是用中文，但又不用中文咭來輔助，究竟爲何能做到這種效果，本文將會由淺入深的爲你剖析。

化學科模擬選擇題練習

爲了讓中五的同學溫習過去所學的，以準備會考，我們由今期開始爲大家提供各科的會考模擬試題練習，首先是化學科。

接連火車卡模擬

這是一個有趣的併字遊戲模擬，利用一輛大車頭在迷宮中穿梭將亂放的英文字拉併成完整的一個英文字，過程非常有趣。

高登最受歡迎遊戲精選

十二月份高登中最受歡迎的一個遊戲軟件，我們選出來供大家娛樂。

撲克遊戲之一排七

排七是紙牌遊戲中最受歡迎的一種。我們將它寫成電腦程序來玩，遊戲的難度及趣味皆大大的提高。

支持老鼠仔的軟件介紹

坊間經已有老鼠出售，價錢不算貴，但目前本港有關支持老鼠仔的軟件很少，我們集集了現時美國所有支持老鼠仔的軟件供各位參考。

SUPPER HELLO

這是一個有趣的HELLO程式，在CATALOG 時，它會自動將各類型的檔案分類，使一目了然。

俄羅斯輪盤

大家不用去葡京，反之安坐家中便可玩輪盤遊戲。這裡最公平不過，你可試試自己的運氣。

35及40軌磁碟檔案地圖顯示器

一般的SECTOR MAP 工具只能用於35軌的磁碟，本工具則能顯示40軌的磁碟檔案地圖。

最新版本蛇之噩夢

電腦時代曾登載過一個遊戲蛇之噩夢，本刊的作者將它再改良，玩時更加精彩緊張。

用來快速COPY 40軌磁碟的 COPY ＋40

我們將COPY＋再修改使之能適用於40軌磁碟COPY。40軌磁碟將成爲一種新趨勢，本工具程式值得收藏。

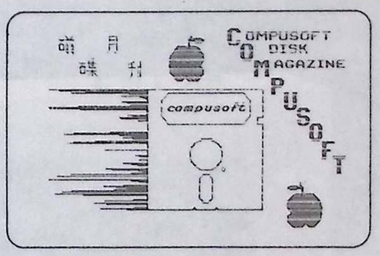
萬用數據處理庫

本數據處理庫能適用於任何情況下，絕對沒有限制，不論應用在那方面，你都會發覺它容易使用。

一九八四年全美四十大最受歡迎遊戲軟件

大家都希望知道有什麼新蘋果軟件出版，這個軟件榜將最新一個月內出版的軟件刊登，大家可清楚了解美國蘋果軟件的發展。

第3期



大逃亡（動畫卡通大賞）

看過大逃亡這個卡通動畫，你會驚覺蘋果電腦在模擬動畫方面的能力，大家可以細心研究這個程式的編寫技巧。

陽曆查陰曆日子修訂版

這個修訂版本，加添了生肖及時辰的顯示，進一步加強本程式的能力。日後將會有一系列的程式應用這個程式，各位萬勿錯過。

你是否樂觀的人——心理測試

這是一個心理測試項目，看看你是否一個樂觀的人。

磁碟雜誌中文字顯示原理剖析㈡

今期繼續講解螢幕顯示中文的原理。希望各位能領悟其中原理，它日能發揮應用。

視窗 HELLO

第二期介紹了一個SUPER HELLO ，用法是將各類型的FILE 分類顯示，但今期這個視窗HELLO 就極之強效，能將上捲的FILENAME 不斷的再顯示，而各項執行方式（RUN、LOAD等）只要按單字母即可。

BUBBLE SORTING 原理剖析及示範

BUBBLE SORTING ：排序在程式編寫中經常會用到，但大家是否清楚了解排序的工作原理？本文提供一個非常有條理的DEMO來詳細解釋BUBBLE 排序的工作原理。導師們最宜用來講解分析給學生聽。

圍棋遊戲

圍棋遊戲是供大家訓練思考力及精神定力的好媒介。各位可與電腦來比試。

愛在深秋

電腦音樂動畫程式——將一首名歌配上美麗的動畫。

第4期

低解像頁繪圖技巧

這篇自學文章詳細解釋了在低解像頁裏繪劃圖案的方法、原理，文章深入淺出，並在文末加上一個具體程式例子，要學習在低解像圖頁繪圖的方法，不能錯過本文。

心理測驗：你是否感受精神壓力

透過回答數十個問題，便可分析閣下現時所受到的精神壓力有多大，是否需要接受心理醫生輔導。

絲綢之路

絲綢之路是一首美妙的音樂，我們的作者將它配上生動的動畫，大家可安坐電腦機前欣賞這首美妙的樂章。

史諾比月曆

這個月曆能印出任何一個年份每月的月曆表，並配上一幅趣緻的史諾比漫畫，擁有印字機的讀者值得收藏。

拼字

這是考驗英文生字字滙能力的一個測驗，你敢接受挑戰嗎？

天降黏泥

這是一個編寫得非常精彩的低解像動畫遊戲，大家在娛樂之餘，不妨參考作者編寫本程式的技巧方法。

電子波子機

一個非常刺激的電子波子機遊戲，未知閣下能否打得一百萬分。

AMPERHELPER

這個工具程式盡量利用了&這個指令，有多個特別簡化的指令，值得大家收藏。

圖書館探險記

這是一個題材非常新穎的ADVENTURE遊戲，作者利用了圖書館編排圖書種類的系統，寫成了這個富有教育意義的遊戲。玩熟這遊戲，閣下在圖書館尋找資料就會快捷輕鬆得多。

電腦辭典

初接觸電腦的朋友，可能對電腦的各方面認識有限，我們特別編寫這個電腦名詞詞滙，將數十個常見的名詞加上註釋，令你對電腦的認識加深。

電子報告板

假如閣下有特別訊息要透過螢幕來告訴大家，不妨利用本程式。

西曆查農曆日子輸入方法之（三）

磁碟雜誌中文字顯示原理剖析（續完）

美國最新出版蘋果軟件

高登軟件流行榜

TAXAN®

超高解像度
彩色電腦螢幕
SUPER VISION III

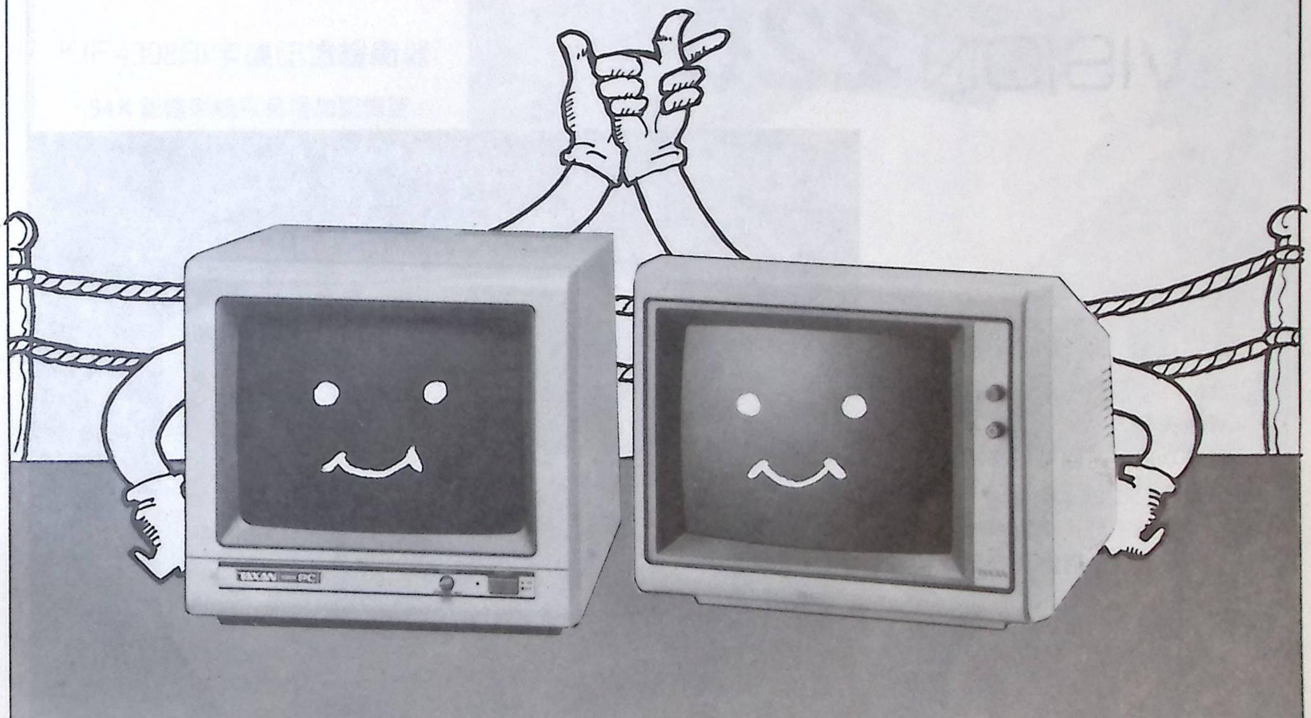


- 獨特三種文字顯示功能，綠色、琥珀色（橙色）及反白色。
- 高解像度 640 點乘 262 行，每點直徑0.37mm，視頻寬達 18 MHz。
- 先進科技，採用SUPER-HIGH-CONTRAST 顯像管，畫面更清晰，久看不倦。
- 採用高品質零件，尤其在顯示彩色圖案時，仍能保持畫面穩定。
- 可另加配附設有石英鐘之底座，調節不同方向、角度，符合人體工程學之要求。
- 完全適用於 IBM PC*、PC/XT*、APPLE II** 及同類型電腦。

總代理：
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司（請認明有限公司，並無分號）
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

* IBM PC、PC/XT is registered trademark of International Business Machine Corp.
** Apple II is a registered trademark of Apple Computer Inc.

TAXAN® 電腦螢幕
與IBM PC*私人電腦
最佳配搭 出類拔萃



TAXAN「KX 1212/1213」單色電腦螢幕及「VISION-PC」RGB彩色螢幕特別爲IBM PC*及同類型電腦而設計，功能及質素均可媲美IBM*原廠產品。

優點特多：

- 特高解像度，視頻特寬。
- 新設計方形平面黑底不反光顯像管，加強畫面反差度，久看不倦。
- KX 單色螢幕備有綠色「長殘影」及琥珀色（橙色）型號，可供選擇。

請駕臨各大分銷商參觀比較。

* IBM及IBM PC爲萬國商業機器註冊商標

享有雲生電腦保証
才是專業性的保用

總代理：
VINCENT COMPUTER CENTRE LTD.
雲生電腦有限公司（請認明有限公司，並無分號）

九龍尖沙咀柯士甸路22-26號好兆年行6字樓 電話：3-7211288-0 電訊：HX 38316 VCC HK

TAXAN®

14吋「混合訊號」彩色電腦螢幕

VISION PAL
VISION 220



- PAL系統（VISION PAL）及NTSC系統（VISION 220）可供選擇。
- 獨特功能，可隨意選擇綠色或琥珀色（橙色）螢幕。
- 內置音响系統，2W輸出功率，適合一般錄影機及多類型電腦。
- 全日本製造。

總代理：

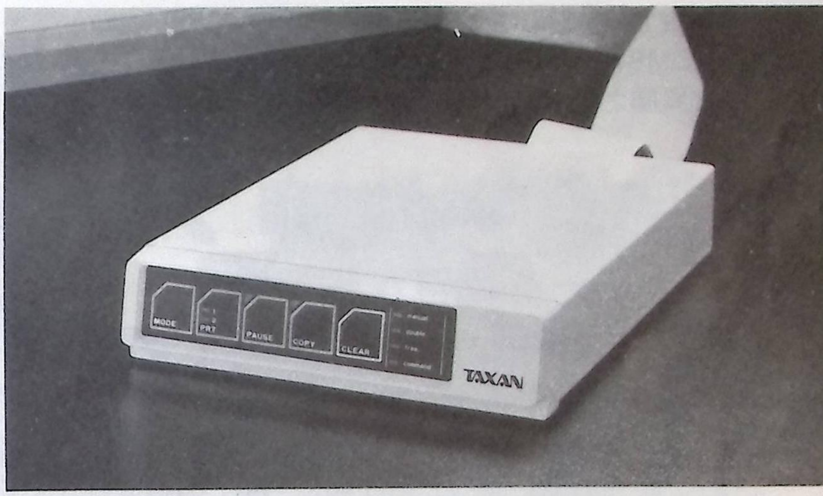
VINCENT COMPUTER CENTRE LTD.

雲生電腦有限公司（請認明有限公司，並無分號）

九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

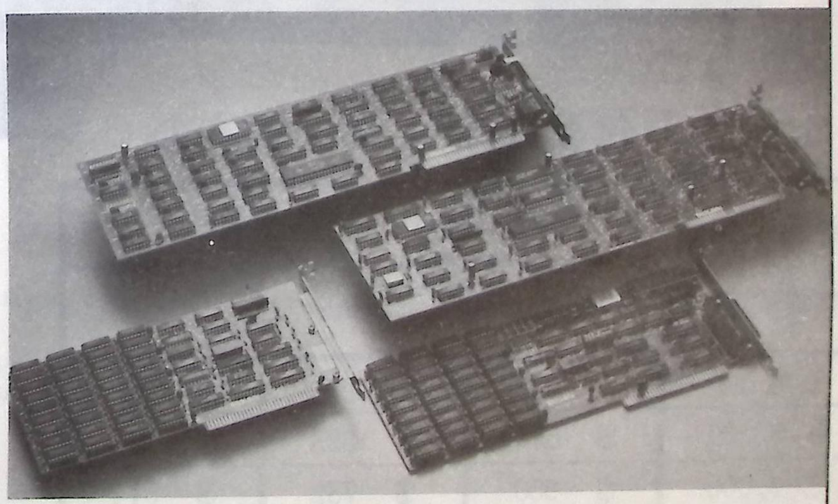
TAXAN®

週邊介面



KIF 4308印字機記憶緩衝器

- 64 K 記憶系統可另增加記憶至256 K。
- 不佔用電腦時間，尤其在印中文字及圖案時。
- 可作無數次COPY。增加工作效率。
- 同時可供兩部印字機使用，可由硬件或軟件控制。



IBM PC*及PC/XT*專用介面

- KIF 3210：256K RAM記憶擴展介面。
- KIF 3210 S：256 K RAM記憶擴展介面及串行式介面接口。
- KIF 3600：彩色/圖案顯示介面。
- KIF 3700：單色顯示及CENTRONICS平行式印字機介面。

總代理：

VINCENT COMPUTER CENTRE LTD.

雲生電腦有限公司（請認明有限公司，並無分號）

九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

*IBM PC,PC/XT are registered trademark of International Business Machine Corp.

現代電腦訓練中心
香港電腦推廣協會

合　辦

全港首創—分期付款購買電腦辦法

可購買 IBM，EAGLE，EVERGO，PC-88，APPLE-IIe，COLUMBIA及 APPLE-II與 IBM之相容機種。亦適合購買上述電腦之MONITOR，DRIVER或PRINTER。



◁分期付款購買電腦計劃表▷

電腦或任何配件之價錢	分期付款每月供款數（免首期）		
	9 mth	12 mth	18 mth
$ 2,000	$ 272	$ 209	$ 147
$ 5,000	$ 679	$ 523	$ 368
$ 7,000	$ 950	$ 732	$ 514
$10,000	$1,358	$1,046	$ 736
$15,000	$2,036	$1,568	$1,104
$20,000	$2,716	$2,092	$1,472
$30,000	$4,072	$3,136	$2,208
$50,000	$6,788	$5,228	$3,680

附帶利益：—

①免費代客選購合用電腦。

②免費安裝。

③低價代客編寫程式。

④免費贈送電腦課程。

⑤免費贈送現成電腦 Program。

⑥一年免費保養。

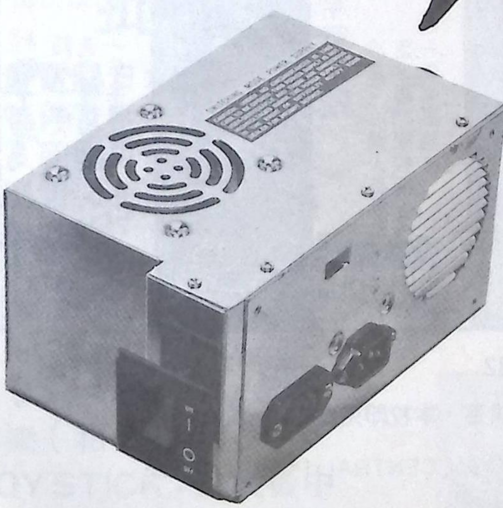
地址：九龍深水埗福華街156號黃金商場地庫20號及54號

電話：　3-601531　3-869085

LEE'S ELECTRONIC CO.

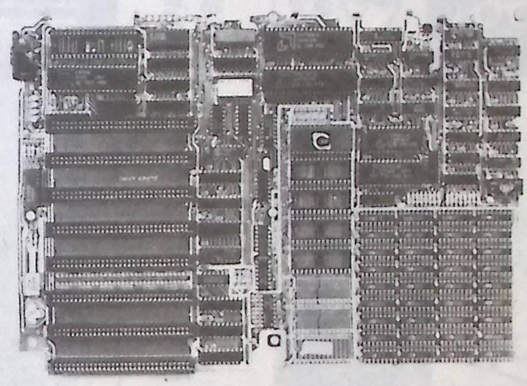
李氏電子公司

130 W



Made in Japan. PC Switching Power Supply

+ 5V, 12A	+ 12V, 5A
− 5V, 0.5A	− 12V, 0.5A



Mother board

* 4.77 MHz 8088 16-bit microprocessor with 8087 co-processor (optional)
* 64K RAM on board with partity check and expandable to 256K
* ROM 8K byte Super XT BIOS expardable to 40K byte
* 4 channel DMA 8237
* 8 channel interrupt 8259
* 8 I/O slot

九龍深水埗黃金商場地庫66A室
電話：3-7291022

Basement Shop No. 66A, Golden Building Shopping Centre, 146-152, Fuk Wah St., Sham Shiu Po, Kowloon.
TEL: 3-7291022

香港北角英皇道255號國都商場3樓
213室　電話：5-718392

Shop No. 213 Olympia Shopping Centre, 2/F., 255 King's Rd., North Point, H.K.
TEL: 5-718392

香港教育司署立案（1957年創校）

世界函授學校

規模大　優點多　科目好

講義
全部經教育司署審定，閱讀容易，印刷精美，免費供應。

導師
國內外著名大學學士，經香港教育司考核合格，教導認真，決不馬虎。

費用
除低廉之學費外，學生決無額外負擔。

文憑
本校頒發文憑，均經政府簽証，實是最佳學歷証件。

校址： 香港九龍彌敦道五六六號9樓A座
電話： ③—803532

初級電腦科
電腦基本原理，電腦操作，邏輯電路，程式編寫八個月畢業。

初級無綫電科
全部圖解教導，原理裝修並重，極易學習，十個月畢業。

半導體裝修科
配合進步的技術，具初學或職業的裝修人員必修的學科十個月畢業。

實用數理
可以加强數理基礎，費用少，收效大，6個月畢業。

x+y=z

實用商科
著重實際應用，加强商業學識，期間短，學習易，六個月畢業。

無綫電工程
設計，研究和深造的理想學科。一年半畢業。

汽車工程
適合職業及業餘人員加强對汽車方面之學識。

冷氣工程
冷氣，冷藏人員缺乏，專門學校極少一年半畢業。

商業美術科
內容新，範圍廣，包括美術和最新廣告設計。一年畢業。

應用漫畫
漫畫可訓練頭腦，增强思想，取材新，學習易，六個月畢業。

INK

本港學生可利用電話詢問索取章程　電話③－803532

請依綫剪下，填好姓名，回郵地址，貼在信封上投寄，本校即免費寄上簡章

郵件請寄：P.O.BOX 70011, KOWLOON, CENTRAL POST OFFICE
九龍中央郵政局信箱 70011 號

技術就是財產

（寄自）
姓名：
地址：
（請註明要學的科目）
- □初級電腦科
- □無綫電工程科
- □初級無綫電科
- □半導體裝修科
- □實用數理科
- □汽車工程
- □冷氣工程
- □實用商科
- □商業美術
- □應用漫畫

請貼上郵票

香港教育司署立案

To：世界函授學校註册處

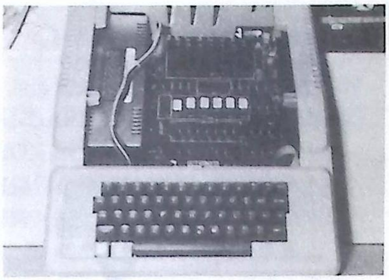
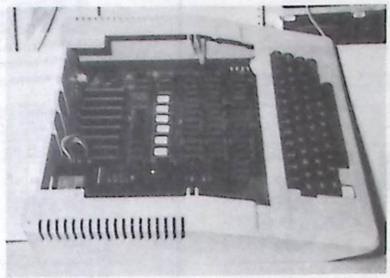
P. O. BOX No. 70011
KOWLOON CENTRAL POST OFFICE,
HONG KONG.

香港九龍中央郵局信箱70011號

電話：K-803532

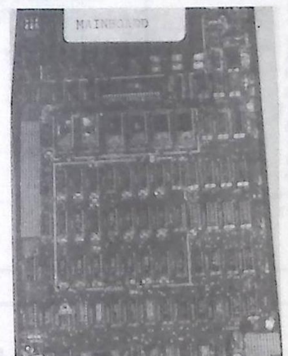
學問就是力量

如發覺自己的電腦出現問題？要立刻修理，否則，會造成嚴重損壞或傷害！但本中心對你的電腦極爲關心，特設維修服務站處理有關電腦問題，有專業人仕解答或修理，如有詢問，請打熱線電話：3- 7783923
＄價錢低廉＄

〈專營〉

多款 電腦硬件，軟件，配件，零件，I.C. 盒帶，磁碟，書籍，保養及維修。



一批發一

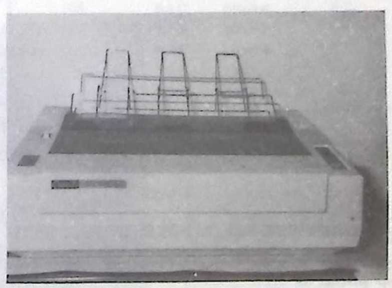
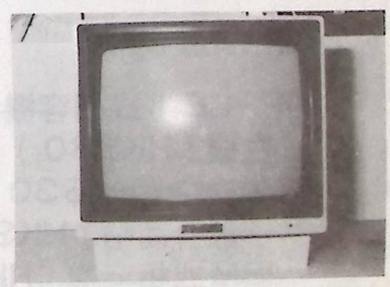
大量電腦日本48K主板，咭件及散料，適合行家及回國外地商戶，低價出售，平靚，有意請來電查詢。

•收購•

二手電腦，任何牌子，磁碟機，顯示器，及其它電腦附件一律歡迎，全港最高，不妨試試。

獨家發售

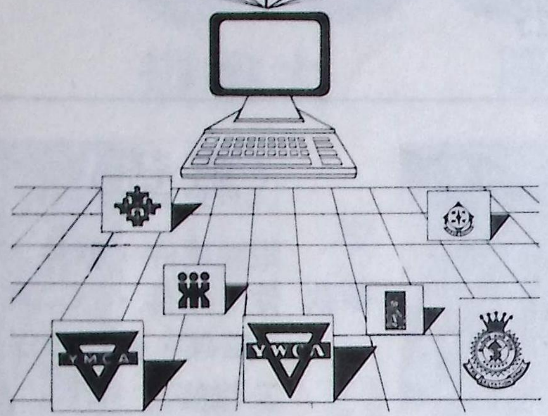
街機（粗暴型）
JOYSTICK，遊戲機中心，一樣刺激緊張，內設强力閃燈，可裝上內藏揚聲器。
只售：199.00

123 電腦中心
COMPUTER CENTRE
電腦維修保養服務站
COMPUTER MAINTENANCE SERVICE

地址：深水埗南昌街白田購物中心一樓 102 室（南昌戲院後面）
電話：3-7783923

電腦證書課程



香港區：	Y.W.C.A.女青年會總會所	5−223101 內線10
	Y.M.C.A.青年會黃傳經中心	5−671571
	香港青年協會民新街中心	5−641277
柴灣區：	循道衛理愛華服務中心	5−580108
灣仔區：	浸會愛羣社會服務處	5−724365
香港仔區：	香港青年協會黃竹坑中心	5−538850
旺角區：	救世軍九龍中央堂	3−312555
深水埗區：	澤安青年中心	3−7763312
黃大仙區：	救世軍修[illegible]隊（業餘進修部）	3−223825
紅磡區：	聖公會聖匠社區中心	3−620301−8
葵涌區：	Y.M.C.A.新界會所	0−200266
荃灣區：	Y.W.C.A.荃灣社會服務處	0−402863
沙田區：	沙角青少年中心	0−6478816
	新翠青年中心	0−6981922
屯門區：	Y.M.C.A.屯門會所	0−848171

IBM 電腦課程

初、中級

· 全期兩個月
· 費用＄380

Y.W.C.A.

· 成績優良者可獲證書

Y.M.C.A.

· 歡迎查詢

中華基督教青年會黃傳經中心
西灣河西河街46號西灣河大廈2樓
5·671571
5·671482

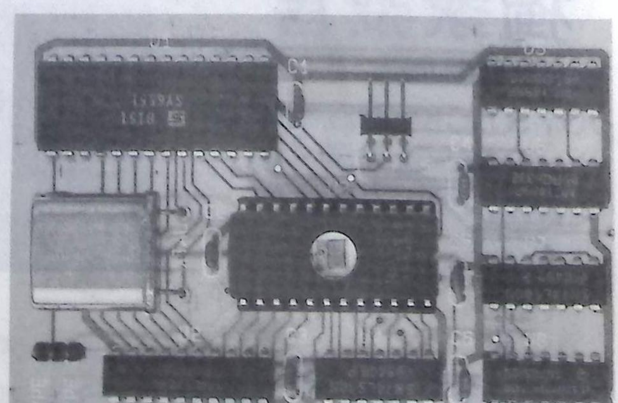
基督教女青年會香港總會
麥當奴道1號
5·223101·10·

專業化電腦硬件維修/改良服務

APPLE II或相容機
—主機（HK＄80）
—磁碟機（HK＄30）
—一般介面咭（HK＄25）
—IBM或相容機（HK＄250）

＊上述費用不包括零件費
＊詳細收費服務表備索

本會擁有一群豐富專業知識的電腦工程人員，抱著非牟利的宗旨為廣大的電腦用家提供廉價而專業化的電腦維修保養／改良及其他服務。



本會現招收基本會員，歡迎參加，章程備索。

電子科技學會 ELECTRONIC TECHNOLOGY ASSOCIATION

專改卡仔

臨時收件站：旺角上海街 438 號同珍商業中心十一字樓四室（電梯十一字）

ASCII 大揭秘

張峰寧

衆所周知，ASC Ⅱ 是AMERICAN STANDARD CODE FOR INFORMA—TION INTERCHANGE 的縮寫。由於電腦只會辨認數目字，故當你輸入一字母時，電腦會將字母轉換爲數字，而ASC Ⅱ 就是負責這工作的。直覺上，ASC Ⅱ 是一個非常簡單的東西，但當你發覺你輸入不同的ASC Ⅱ 值，而得出相同的結果時，你的觀感會180 度轉變。

就以字母〝A〞來舉例，當你打入PRINT CHR＄（65）或PRINT CHR＄（193） 都會得出〝A〞字。但當你將此倆加以比較，他們彷彿是不同的，現試打入PRINT CHR＄（65）＝CHR（193），此時螢幕會出現〝0〞。〝0〞是代表電腦認爲CHR＄（65）＝CHR＄（193）這個STATEMENT 是不相等的。（如果出現〝1〞則表示此STATEMENT 是對的。）爲什麼呢？明明兩個都出現〝A〞，但爲什麼電腦會說不相等呢？

現在，讓我們深入探討這問題。A—SC Ⅱ 將0—127 間128個值用於控制字母及符號（參考表一）。如果你在C—HR＄（X）中，設X值爲0—127 其中一數，則會得出相對的字母或符號。

另外，你也可使用128—255 之值，而ASC Ⅱ7與135都會產生〝BEEP〞一聲（CTRL—G）。換句話說，兩者之不同是位元組（BITS）及字節（BYT—ES）之階層。

現讓我們看看127 這數目，你會發覺127 是一個字節裏頭七個位元組所能產生之最大數。（參考表二）現將表2中BIT VALUES 之值相加，便會得出127 ，或在二進制下之01111111。故此，我們知道，ASC Ⅱ 碼之轉換時，是只用了頭7 個BIT來工作，而第八個則是空的。

現再以〝A〞爲例，如我們假設A—SC Ⅱ 值爲65，其二進制則是01000001，但如果設定X爲193 時，其二進制是11000001。而我們發覺在ASC Ⅱ 193 時，第八個BIT是存放着東西的。其值等於128（64×2）＋65＝193。

Decimal	Hex	Key	Tradit. Name(1)	Function
0	$00	CTRL-@	NULL	
1	$01	CTRL-A	SOH	
2	$02	CTRL-B	STX	
3	$03	CTRL-C	ETX	Stop Running Program
4	$04	CTRL-D	ET	DOS Flag
5	$05	CTRL-E	ENQ	
6	$06	CTRL-F	ACK	
7	$07	CTRL-G	BEL	Bell
8	$08	CTRL-H	BS	Backspace,Left Arrow
9	$09	CTRL-I	HT	Tab
10	$0A	CTRL-J	LF	Linefeed, Down Arrow(IIe)
11	$0B	CTRL-K	VT	Up Arrow (IIe)
12	$0C	CTRL-L	FF	Form Feed
13	$0D	CTRL-M	CR	RETURN
14	$0E	CTRL-N	SO	
15	$0F	CTRL-O	SI	
16	$10	CTRL-P	DLE	
17	$11	CTRL-Q	DC1	
18	$12	CTRL-R	DC2	
19	$13	CTRL-S	DC3	Stop listing program
20	$14	CTRL-T	DC4	
21	$15	CTRL-U	NAK	Right Arrow
22	$16	CTRL-V	SYN	
23	$17	CTRL-W	ETB	
24	$18	CTRL-X	CAN	Cancel Line
25	$19	CTRL-Y	EM	
26	$1A	CTRL-Z	SUB	
27	$1B	ESC	ESC	
28	$1C	None	FS	
29	$1D	None	GS	
30	$1E	None	RS	
31	$1F	None	US	
32	$20	SPACE		

表(1) 標準ASC Ⅱ字符代碼

因此，你會發現，在BASIC 下，字母轉換所用之ASC Ⅱ 值是0—127，但在電腦本身，爲方便起見是用128—255 之值。現試鍵入程式一。

```
100 HOME
120 FORI=0 TO 127
130 PRINT CHR$(I);
140 NEXT
```

程式(1)

此程式之進行結果在APPLE 各類型機上都不同。在APPLE Ⅱ及APPLE Ⅱ十下，此程式之結果是不會印出小階字母的。但在Ⅱe 上則會。

但，如果你將程式一之行號120 改爲FOR I—128 TO 255 ，結果也是一樣。APPLE只是將ASC Ⅱ值加 128而已。

讀者可嘗試用PRINTER 印出此結果，但你會發現這時APPLE，APPLE Ⅱ十及Ⅱe 也會印出小階。因爲，在此類型電腦主機上是沒有小階字母的，但你的印字機是有的。（當然，冒牌貨是例外。）

字體

爲什麼蘋果本身要用128—255 的ASC Ⅱ 值去做轉換工作呢？

事實上，螢幕的影像是受硬件控制的。當蘋果輸送一條位元組去電視輸出時，BIT是會顯示於螢幕上的。如BIT是 0 的話，則表示空白，如 1 則用一個小點代表。一個空白的螢幕表示所有輸送到螢幕的位元組都是 0 ，而訊號輸去螢幕之最濶長度，是280 點，而高度則是192 ，故在高解像圖形下，螢幕之極

限是280×192。

但蘋果是如何知道什麽樣的BITS是被送來呢？此問題與其畫面地址結構有關。在一個文字幕下有1024個地址，但其中只有960 個可供使用。此數是因爲文字幕正常乃長40寬24。而每一點代表1 BYTE的記憶。電腦就是讀取這些資料從而知道要顯示些什麽出來。

每個位置是佔去56點的。（因爲280／40＝7，192／24 ＝8，7×8＝56）故每一個螢幕上的字元是用56個分開的BITS 所造成的。但是一個BYTE才等於8 BITS ，爲何文字幕一個BY—TE 等於56個BIT S 呢？此問題是有關FONT 的。

FONT 是存放在APPLE ROM 中一大組的BYTE，每8個BYTE便造出一字。現以“A”爲例解釋之：

各位曾否看過“A”的圖形？（見程式二）此圖形是横有8行，直有7行，而是放在5×7的矩陣內。

由於要建立一字，須有8 BYTE ，而每個BYTE要7個BITS ，故蘋果會尋找記憶體中的一個8 BYTE的位置來顯示一字，當然，如果小階圖形是沒有被放在記憶中的話，則電腦不會印出。而在AP. II及II＋F，（表三）中之形式是不被存於在記憶的，當然也是不會被顯示的。

但在AP IIe 上，表三中之形式是被存放在記憶中的，當然是會被顯示的。

Decimal	Hex	Key	Decimal	Hex	Key
33	$21	!	81	$51	Q
34	$22	"	82	$52	R
35	$23	#	83	$53	S
36	$24	$	84	$54	T
37	$25	%	85	$55	U
38	$26	&	86	$56	V
39	$27	'	87	$57	W
40	$28	(	88	$58	X
41	$29	)	89	$59	Y
42	$2A	*	90	$5A	Z
43	$2B	+	91	$5B	[
44	$2C	,	92	$5C	\
45	$2D	-	93	$5D	]
46	$2E	.	94	$5E	^
47	$2F	/	95	$5F	_
48	$30	0	96	$60	space(2)
49	$31	1	97	$61	a
50	$32	2	98	$62	b
51	$33	3	99	$63	c
52	$34	4	100	$64	d
53	$35	5	101	$65	e
54	$36	6	102	$66	f
55	$37	7	103	$67	g
56	$38	8	104	$68	h
57	$39	9	105	$69	i
58	$3A	:	106	$6A	j
59	$3B	;	107	$6B	k
60	$3C	<	108	$6C	l
61	$3D	=	109	$6D	m
62	$3E	>	110	$6E	n
63	$3F	?	111	$6F	o
64	$40	@	112	$70	p
65	$41	A	113	$71	q
66	$42	B	114	$72	r
67	$43	C	115	$73	s
68	$44	D	116	$74	t
69	$45	E	117	$75	u
70	$46	F	118	$76	v
71	$47	G	119	$77	w
72	$48	H	120	$78	x
73	$49	I	121	$79	y
74	$4A	J	122	$7A	z
75	$4B	K	123	$7B	{
76	$4C	L	124	$7C	|
77	$4D	M	125	$7D	}
78	$4E	N	126	$7E	~
79	$4F	O	127	$7F	Delete
80	$50	P			

文字頁和圖像頁的分別

既然文字被印在螢幕上之組織是與高解像圖形一樣，爲何我們不能在高解像頁上，直接印出文字呢？原因很簡單，因高解像頁是用BIT—MAPPED的，而文字幕不是。況且，在高解像下要顯示一東西的話，是需要用文字幕8倍的記憶體。故要在高解像下印出文字，你必須小心地將每個字逐行逐點地CO—DE上。

在高解像下，1個BYTE在螢幕上是有7個DOTS的（第8個BIT不是被顯示的；其乃爲COLOR所用）。而一條横的DOT—LINE是280點，280／7＝40，故一處非常細小的螢幕部份需要40個BYTE，而我們有192行，故40×192＝7680，7680個BYTE是被用到去畫幅圖形的。

螢幕代碼

好了！現在轉入正題，讓我們談談SCREEN ASC II 。字元可以在正常、反白、閃動三種模式下顯示。

Bit Number	8	7	6	5	4	3	2	1
Bit Value	0	64+32+16+8+4+2+1						

表(2)

蘋果電腦是用不同的ASC II 碼去分辨要印出的字元是正常？反白？或閃動的？

```
0000000
0000000
0000000
0011000
0000100
0011100
0100100
0011010
```

表(3)

i)　0～63：是用作反白的，而第7，8個BITS 大多是被設爲0。

ii)　64～127 ：是用作閃動顯示，第8個BITS 是設爲0的。

iii）128～255 ：是用作正常顯示的，第7，8個BITS 是通常不被設爲0的。

請注意：現在ASC Ⅱ 碼0～127是被分爲兩份，本來是CTRL字元用的代碼是被反白字元代替了。而本來大階的代碼，現被變爲閃動的代碼，而在小階下，我們更有一些符號（參考表4）。而由於CTRL一字元及小階之缺少，我們可以在7個BITS 中，得出二組符號——反白及閃動。

第三種碼是用作正常顯示的，CTRL 字元128—159 是不會出現的。

但記着，這些都是螢幕代碼，故當你打入PRINT CHR＄（1） 是不會出現一個反白〝A〞的。你必須打入PRINT INVERSE：PRINT CHR＄（65）。有關其關係。請參考表五。

反白與閃動原理

現在我們要談談反白及閃動之形式方法。首先談反白之形成。爲使各位更爲明白，我們再以〝A〞圖形爲例，反白只不過是將BITS 由0變爲1，1變爲0，此項工作電腦是毋須要用一個FONT，當INVERSE 命令被用，它只須FLIP 去那影象卽可。

而閃動之形成只不過是正常及反白形式的快速轉換畫而已，當FLASH 命令被用，INVERSE及NORMAL 便會互相交替，並顯示於螢幕上。

好了！希望各位在讀完此文後，能對ASC Ⅱ 有更深入的認識。另外本人編寫了一個名爲ASC & CHR＄LIST的程式，以供讀者使用。（程式三）■

Screen Code Number	Apple II,II+	Apple IIe Primary Set	Apple IIe Alternate Set
0-31	Uppercase Inverse	Uppcarse Inverse	Uppercase Inverse
32-63	Special Inverse	Special Inverse	Special Inverse
64-95	Uppercase Flash	Uppercase Flash	Uppercase Inverse
96-127	Special Flash	Special Flash	Lowercase Inverse
128-159	Control Charact.	Uppercase Normal	Uppercase Normal
160-191	Special Normal	Special Normal	Spesial Normal
192-223	Uppercase Normal	Uppercase Normal	Uppercase Normal
224-255	Special Normal	Lowercase Normal	Lowercase Normal
Notes:	No lowercaase at all unless specially equipped	No lowercase inverse or flashing modes.	No Flash in any mode. Gives Inverse lowercase

'Special' means typographical marka, math operators ,and the numerals.

表(4)APPLE Ⅱ，Ⅱ十及Ⅱe的螢幕代碼

Command	Screen Memory		On Screen
PRINT CHR$(65)	$C1	(193)	A
PRINT CHR$(193)	$C1	(193)	A
INVERSE:PRINT CHR$(65)	$01	(1)	Inverted A
INVERSE:PRINT CHR$(193)	$01	(1)	Inverted A
FLASH: PRINT CHR$(65)	$41	(65)	Flashing A
FLASH:PRINT CHR$(193)	$41	(65)	Flashing A
NORMAL: PRINT"A"	$C1	(193)	A

表(5)螢幕代碼與ASC Ⅱ碼的關係

```
100  REM     HI RES A'S
102 :
104  REM     DEMONSTRATES HOW TO
106  REM     MAKE A SMALL LETTER
108  REM     ON THE HIGH RES
110  REM     SCREEN
112 :
200  HGR
210 P = 8192
220  FOR K = 1 TO 2
240  FOR I = 0 TO 7168 STEP 1024
250  READ N
260  POKE P + I,N
270  NEXT I
280 P = P + 1
290  NEXT K
300  RESTORE
310  IF P < 8192 + 40 THEN 220
320  END
330 :
340  DATA  0,0,0,12,16,28,18,44
350  DATA   0,8,20,34,34,62,34,34
360 :
370  REM    8192 IS START OF
380  REM    THE HI-RES SCREEN.
390  REM    EACH LINE OF DOTS
400  REM    FORMING A LETTER
410  REM    IS 8192 +1024
420  REM    MEMORY  LOCATIONS
430  REM    APART.
440  REM
450  REM    IN LINE 280, 8192
460  REM    IS INCREMENTED SO
470  REM    THAT THE NEXT
480  REM    CHARACTER CAN BE
490  REM    PRINTED. THIS IS
500  REM    DONE 40 TIMES FOR
510  REM    ONE LINE OF
520  REM    LETTERS.
530  REM
540  REM    DATA IN LINE 340
550  REM    MAKE THE SMALL
560  REM    A. LINE 350 FORMS
570  REM    THE CAPITAL A.
580  REM    THE LETTERS MUST BE
590  REM    'BACKWARDS' TO COME
600  REM    OUT RIGHT!!
```

程式(2)在高解像頁裏產生〝A〞

```
4   REM  *************************
5   REM  ** PROGRAM OF  LISTING **
6   REM  **      ASCII & CHR$   **
7   REM  **    BY TONY CHANG    **
8   REM  *************************
10   TEXT : HOME
20   VTAB 4: PRINT "What do you wa
      nt to find ?"
30   VTAB 10: HTAB 10: PRINT "<1>
      ASCII"
40   VTAB 12: HTAB 10: PRINT "<2>
      CHR$(N)"
45   VTAB 14: HTAB 10: PRINT "<3>
      QUIT"
50   VTAB 16: HTAB 15: PRINT " You
      r choice  ";: INPUT A
60   IF A = 1 THEN 100
70   IF A = 2 THEN 200
72   IF A = 3 THEN  HOME : PRINT "
      GOODBYE!!!": END
75   IF A < 1 THEN 10
76   IF A > 3 THEN 10
100  HOME : VTAB 5: INPUT "To fin
      d ASCII, enter the letter: "
      ;A$
110  VTAB 8: HTAB 5: PRINT "ASC("
      ;A$;")="; ASC (A$)
120  GOSUB 225
200  HOME
210  VTAB 5: HTAB 5: INPUT " Ente
      r N:";N
220  VTAB 10: HTAB 5: PRINT "CHR$
      (";N;")="; CHR$ (N)
222  GOSUB 225
225  VTAB 20: HTAB 5: PRINT "PRES
      S <M> TO MENU ...";
226  INPUT C$
227  IF C$ <  > "M" THEN 225
228  IF C$ = "M" THEN 10
229  RETURN
```

程式(3)ASC Ⅱ& CHR＄互換表
此程式是用來找對每個字符相對的ASC Ⅱ值

學習迷宮產生的原理及自編迷宮遊戲

WONG LEUNG FUNG

相信各位也曾看過或玩過迷宮式的電子遊戲，但你們却會否嘗試自行編寫一個迷宮遊戲程序呢？以下便有一個實用的迷宮產生器程式，分別以APPLE SOFT BASIC和6502機械碼（2大版本）所編寫，好讓各讀者能比較兩者的速度和效能之差距。

利用電腦去產生一個縱橫交錯的迷宮，其實也不是一件困難的事情，這祇要大家能坐下來和動動你的腦筋便可了！在衆多由BASIC 所編寫的迷宮產生器中，其原理和效果也大致相同，是在一段記憶體內產生一個迷宮和絕不容許在迷宮有一些困死的路徑或未塡上迷徑的空間存在。

當然，一些比較複雜的迷宮，往往會有很多循環的路徑，而使玩者迷失其方向。言則，一個巨大而複雜的迷宮，其產生及運算過程是往往可能要花費10～20分鐘的時間呢！這可算是大煞興緻了，所以筆者特別以6502 機械碼編寫了一個迷宮產生器副程式，來幫助各同好去編寫一個完整和有趣的迷宮程式。

字幕式迷宮

整個機械碼程式，亦祇佔用了834 BYTES的記憶空間，而且程式是擺放於APPLE的＄7000位址高記憶地址上去。祇要用者按方法去呼喚（如CALL 28685— 去執行整個迷宮產生程序），程序便會發揮其效能。在APPLE 記憶體＄6000起建立了一個39格×23行的迷宮陣，而剛巧適合AP—PLE 40字×24行的文字幕顯示格式（參看圖1）。

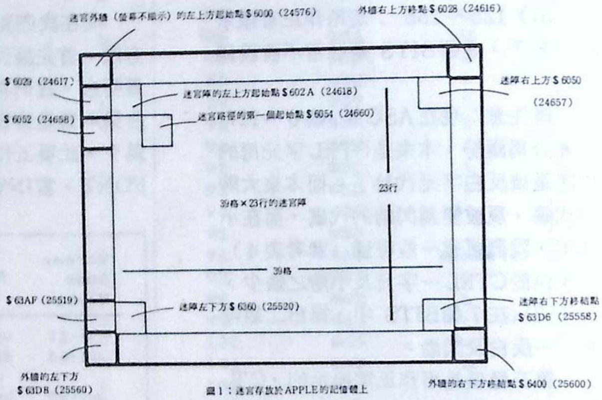


圖1：迷宮存放於APPLE的記憶體上

還有，在這834 BYTES的程式中，迷宮產生器亦祇佔用了3份1的長度，其餘的部份便是將迷宮顯示和在迷宮放入老鼠仔去尋找出路等副程式。

```
10  REM  **************************
     ****
20  REM  * Applesoft Maze generat
     or *
30  REM  *     It is slower than
     *
40  REM  *   Machine language for
     *
50  REM  *      at least 100 %
     *
60  REM  * Programed by Lawson Wo
     ng *
70  REM  **************************
     ****
80  HOME : DIM M(1025)
90 A(0) = 2:A(1) =  - 82:A(2) =  -
    2:A(3) = 82:W = 255:L = 32:S
    T = 85
100  FOR I = 1 TO 1025:M(I) = 90:
     NEXT
110 X = 43: FOR I = 1 TO 23
120  FOR Y = 0 TO 38
130 M(X + Y) = 255: NEXT Y
140 X = X + 41: NEXT I
150 M(ST) = 69
160 J =  INT ( RND (1) * 4):H = J
170 B = ST + A(J): IF M(B) = W THEN
     M(B) = J + 65:M(ST + A(J) /
     2) = L:ST = B: GOTO 160
180 J = (J + 1) * (J < 3): IF J <
     > H THEN 170
190 J = M(ST) - 65:M(ST) = L: IF
     J < 4 THEN ST = ST - A(J): GOTO
     160
200  GOSUB 250
210 M(ST) = 64:J = 3
220 B = ST + A(J) / 2: IF M(B) =
     L THEN M(B) = 64:M(ST) = L:S
     T = B:J = (J + 2) + 4 * - (
     J > 1)
230 J = (J - 1) + 4 * (J = 0): GOSUB
     250: GOTO 220
240  END
250 X = 43
260  FOR Z = 1 TO 23:A$ = "": FOR
     I = 0 TO 38
270 A$ = A$ +  CHR$ (M(X + I))
280  NEXT I
290  VTAB Z: HTAB 1: PRINT A$:X =
     X + 41
300  NEXT Z
310  RETURN
```

程式1：以BASIC寫成的迷宮產生器

學習迷宮產生的原理及自動編迷宮遊戲

換言之，程式是可分爲多個不同副程序和進入點，用者可按不同的需要而呼喚不同的副程式去執行不同的程序。爲了使程式具有流動擺放和靈活性，程式內大部份的絕對值和參數值均以滙編語言的標記（LABEL)所代表，所以用者祇須在原程式（ASSEMBLE SOURCE CODE）上改變ORG＄700D（原程式碼在滙編成機械碼後擺放於＄700D爲起始的指令）及其他參數LABEL內容，則整個副程式便可任意擺放於64K記憶中的任何一處了。

同樣，程式也以PASCAL 結構語言的邏輯去編寫，每個副程式在未執行前是需要置定某些參數值，它們才可發揮相應的效能。所以置定不同的參數，便可有同樣的程序但却不同的輸出效果呢！如用者將此機械碼（程式3）讀入電腦後，再執行CALL 28685時，電腦便會將＄6054至＄63D6的迷宮圖案記憶內容原本地輸出文字幕上（參看圖1），接着老鼠便會在迷宮中走動。若用者按SPACE BAR 鍵，則電腦便會再產生另一幅迷宮；反之按ESL 鍵，程式便會中止及跳到APPLE的MONITOR程式上去。

還有，在我們未執行CALL 28787 指令前，那不妨嘗試輸入POKE 28796，0：POKE 28801，9×256 ：POKE 28710，60：CALL 28787 。這樣，你可發覺電腦會將＄900至＄0CAC 的記憶內容顯示出螢幕，並跳回 BASIC 的控制系統中。言則，這便可是一個記憶體窗戶（WINDOW OF MEMORY）顯示程式了。

整個迷宮產生器的滙編碼程式（ASSEMBLE SOURCE LISTING）亦已收錄於〔電腦時代〕第16期的〔程式磁碟〕中，祇要用者以APPLE LISA 滙編語言系統讀入原程式，再按A（代表ASSEMBLE THE SOURCE CODE）將原碼變爲代碼（亦即機械碼），然後按CRTL—D和鍵入BSAVE MAZE（OBJECT），A＄800，L＄342按RETURN 便可。但必須緊記在重新讀入代碼於記憶體去執行時，其過程必爲BLOAD MAZE（OBJECT），A＄700D才可。

以上均是針對一些對6502 機械碼有認識和懂得操作一APPLE LISA 的同好而言。那麼一些BASIC 發燒友又怎樣辦呢？可能這個機械碼迷宮產生器亦只是一件缺乏參考資料或可讀性的工具了，又或對他們自行編寫迷宮程式帶來恐懼和障礙。

因此，筆者亦以同一原理和邏輯編寫了一個APPLESOFT 的迷宮產生器（程式1），與及一個利用機械碼迷宮產生器輔助的立體迷宮樣本（程式2），好讓各同好能揣摸、吸收和比較BASIC與機械碼程序效率的分別了。

迷宮架構

參閱程式1，在迷宮未產生之前我們是必須將迷宮內的場地清除（載入CHR＄（255）），並在迷宮四周加上隱形（不顯示於螢幕）圍牆（載入CHR＄（90））。

在以上的準備工夫做妥後，下一步電腦便會先在迷宮指定的左上角爲起始點（列陣M（85））爲起始點，並放下「E」爲起始點標記，然後分別以隨機數字0，1，2，3 爲右

程式2：配合機械碼程式迷宮產生器的立體迷宮

```
10  REM  ************************
20  REM  *   3-D Maze Sample   *
30  REM  * Maze Generator(MC) *
40  REM  *      Program by     *
50  REM  * Mr.Wong leung fung *
60  REM  ************************
70  REM  **INIT OF PARAMETER **
80  IF  PEEK (28685) = 169 THEN 1
    00
90  PRINT  CHR$ (4);"BLOAD MAZE(O
    BJECT)"
100 S1 = 28685:S2 = 28787:RTS = 2
    8710
110 A =  INT ( RND (1) * 256): POKE
    78,A:A =  INT ( RND (1) * 25
    6): POKE 79,A
120 POKE RTS,96
130 CALL S1
140 FLASH : VTAB 24: HTAB 2: PRINT
    "YOU ARE INSIDE MAZE, TRY ES
    CAPE NOW !"; : NORMAL
150 ST = 24660:F = 41:L = 1:R =  -
    1
160 POKE ST,192: CALL S2
170 REM  ** MAIN LOOP **
180 GOSUB 900
190 IF K = 205 THEN  TEXT : GET
    K$
200 IF K = 160 THEN 300
210 IF K = 149 AND F = 41 THEN F
     =  - 1:L = 41:R =  - 41: GOTO
    330
220 IF K = 149 AND F =  - 1 THEN
    F =  - 41:L =  - 1:R = 1: GOTO
    330
230 IF K = 149 AND F =  - 41 THEN
    F = 1:L =  - 41:R = 41: GOTO
    330
240 IF K = 149 AND F = 1 THEN F =
    41:L = 1:R =  - 1: GOTO 330
250 IF K = 136 AND F = 41 THEN F
     = 1:L =  - 41:R = 41: GOTO
    330
260 IF K = 136 AND F = 1 THEN F =
     - 41:L =  - 1:R = 1: GOTO 3
    30
270 IF K = 136 AND F =  - 41 THEN
    F =  - 1:L = 41:R =  - 41: GOTO
    330
280 IF K = 136 AND F =  - 1 THEN
    F = 41:L = 1:R =  - 1
290 GOTO 330
300 IF  PEEK (ST + F) = 255 THEN
     PRINT  CHR$ (7);: GOSUB 920
    : GOTO 330
310 POKE ST,160:ST = ST + F: POKE
    ST,192
320 CALL S2
330 REM  CHANGE DIRECTION
340 TT = ST + F
350 FOR X = 0 TO 2: FOR Y = 0 TO
    2:A(X,Y) = 0: NEXT Y,X
```

、上、左、下的方向間格測試是否可行（間格內的數值為一255，則為可行；反之不可行）。

假設是可行，則電腦首先在跳往之處放下去向的標記，並在原坐標與跳往坐標間的列陣置入CHR$（32）以貫通迷徑；反之，去路有阻，則將剛才獲得的隨機數（0，1，2或3）為首依序循環測試（如隨機數為1變2、2變3、3變0、0變1），直至一個循環亦未能找到可行之處。這亦表示現在坐標的前、左和右方向均為絕路，所以電腦便會拾回原駐坐標上的去向標作反方向而行，繼之再隨機尋覓其他可行之路。

最後，迷宮場內的每一部份必定能填滿迷徑，而電腦亦會循原路走回原始的起步點，也即意味迷宮產生過程完畢。

在程式1中，迷宮產生的每一個步驟亦已清楚顯示於螢幕上，相信各位亦能了解其原理和邏輯。程式中，作者還加上了老鼠在迷宮尋找出路的過程，其原理也是依照左手定律一老鼠仔永遠是先向左方而行，若迷宮是沒有循環的路徑，老鼠仔是可走遍迷宮的每一個角落。

比較程式1與機械碼迷宮產生器，其效果和原理一致，但兩者在速度上則天淵之別！或許你會投訴為什麼以數行的BASIC指令句便可達到迷宮的產生，而機械碼則需834 BYTE那麼長呢？

答案也非常簡單，第一，以BASIC高階語言去編寫程式，往往比以低階語言為快，但却失掉了速度和效率。第二，你可不要忘記BASIC程式的背後，是隱藏了10幾K的APPLE SOFT BASIC ROM所支援呢！比方細心計算單一個BASIC中的HOME（清除畫面）指令，其所需的機械碼指令加上一BASIC文字指令格式，其長度可能是需要幾百個BYTES和步驟呢！

立體迷宮

惟恐各讀者未能完全了解和應用機械碼迷宮產生副程式，筆者亦精心編寫了一個立體迷宮的樣本（程式2），好使各讀者能置身於迷宮之中。程式中，電腦是會讀入機械碼副程式，並顯示整個迷宮的鳥瞰圖及你之所在處，按左右箭咀則為改變自己面對的方向，而SPACE BAR是代表向前行一步。

程式是只容許玩者在迷宮中前看3步的景象，而當玩者面向牆壁時，螢幕是會完全反白。若再按SPACE BAR則電腦會發出「嗶」一聲，以表示撞牆的意思。在迷失方向時，讀者可按「M」鍵作鳥瞰迷宮圖，而鳥瞰次數不限。

接着再重談機械碼迷宮產生器副程式，其SOURCE CODE已收錄於〔電腦時代〕的〔程式磁碟〕集內。當然，你必須對6502 機械碼有所認識，這才可着手改良和參考此副程式。

反之，這也不是一個很大的問題，因為迷宮產生器的機械碼已詳列於程式3中，鍵入方法請參閱APPLE 應用手冊，或參閱電腦時代第三期〔如何打入機械碼〕。還有，若你們想進一步了解機械碼的迷宮產生過程，那不妨在CALL

程式3：機械碼迷宮產生器

```
************************
** MACHINE LANGUAGE **
**  MAZE GENERATOR  **
**  A$700D, L$335   **
************************
700D- A9 FF 85
7010- 32 20 33 FB 20 58 FC 20
7018- 39 70 20 54 70 20 73 70
7020- 20 8D 70 20 73 70 20 50
7028- 71 C9 9B F0 06 20 1B FD
7030- 4C 17 70 20 58 FC 20 69
7038- FF A0 00 84 18 8C 07 70
7040- A2 60 86 19 A2 29 8E 06
7048- 70 8E 08 70 A2 19 A9 EE
7050- 20 09 72 60 A0 2A 84 18
7058- A0 60 84 19 A0 29 8C 08
7060- 70 A0 00 8C 07 70 A2 27
7068- 8E 06 70 A2 17 A9 FF 20
7070- 09 72 60 A0 00 84 24 84
7078- 25 8C 07 70 A2 2A 86 18
7080- A2 60 86 19 A2 17 20 22
7088- FC 20 30 72 60 A9 54 85
7090- 18 8D 09 70 A9 60 85 19
7098- 8D 0A 70 A9 C5 A0 00 91
70A0- 18 20 AD 72 8D 00 70 8D
70A8- 01 70 20 5F 72 18 AC 09
70B0- 70 84 18 AC 0A 70 84 19
70B8- A5 18 6D 04 70 85 18 A5
70C0- 19 6D 05 70 85 19 A0 00
70C8- B1 18 C9 FF D0 2D AD 01
70D0- 70 18 69 C1 91 18 18 A6
70D8- 18 AD 09 70 8E 09 70 6D
70E0- 02 70 85 18 A6 19 AD 0A
70E8- 70 8E 0A 70 6D 03 70 85
70F0- 19 A9 A0 91 18 18 18 18
70F8- 4C A1 70 AD 09 70 85 18
7100- AD 0A 70 85 19 AD 01 70
7108- C9 03 90 10 A9 00 8D 01
7110- 70 AD 01 70 CD 00 70 F0
7118- 09 4C AA 70 EE 01 70 4C
7120- 11 71 B1 18 48 A9 A0 91
7128- 18 68 38 E9 C1 C9 04 F0
7130- 1E 20 5F 72 38 A5 18 ED
7138- 04 70 8D 09 70 85 18 A5
7140- 19 ED 05 70 8D 0A 70 85
7148- 19 18 18 18 4C A1 70 60
7150- A9 C0 A0 54 84 18 8C 09
7158- 70 A0 60 84 19 8C 0A 70
7160- A0 00 91 18 A9 02 8D 01
7168- 70 8D 00 70 A0 00 AD 01
7170- 70 20 5F 72 20 39 73 C9
7178- A0 F0 41 C9 9B F0 3D AD
7180- 09 70 85 18 AD 0A 70 85
7188- 19 18 A5 18 6D 02 70 85
7190- 18 A5 19 6D 03 70 85 19
7198- B1 18 C9 A0 F0 1F AD 01
71A0- 70 A2 00 8E 01 70 C9 00
71A8- D0 05 A9 04 8D 01 70 AD
71B0- 00 70 38 E9 01 18 6D 01
71B8- 70 4C 66 71 60 A9 C0 91
71C0- 18 A6 18 AD 09 70 8E 09
71C8- 70 85 18 A6 19 AD 0A 70
71D0- 8E 0A 70 85 19 A9 A0 A0
71D8- 00 91 18 20 73 70 AD 01
71E0- 70 C9 01 90 04 F0 02 B0
71E8- 18 A9 00 8D 01 70 AD 00
71F0- 70 18 69 02 38 ED 01 70
71F8- 8D 00 70 8D 01 70 4C 9E
7200- 71 A9 04 8D 01 70 4C EE
7208- 71 91 18 C8 CC 06 70 F0
7210- 03 4C 09 72 EE 07 70 EC
7218- 07 70 F0 13 48 A5 18 18
7220- 6D 08 70 90 02 E6 19 85
7228- 18 68 A0 00 4C 09 72 60
7230- B1 18 C8 20 ED FD CC 06
7238- 70 F0 03 4C 30 72 EE 07
7240- 70 EC 07 70 F0 18 A5 18
7248- 18 6D 08 70 90 02 E6 19
7250- 85 18 A0 00 84 24 E6 25
7258- 20 22 FC 4C 30 72 60 A2
7260- 00 8E 03 70 8E 05 70 C9
7268- 00 F0 13 C9 01 F0 1A C9
7270- 02 F0 29 A2 29 8E 02 70
7278- A2 52 8E 04 70 60 A2 01
7280- 8E 02 70 A2 02 8E 04 70
7288- 60 A2 D7 8E 02 70 A2 FF
7290- 8E 03 70 8E 05 70 A2 AE
7298- 8E 04 70 60 A2 FF 8E 02
72A0- 70 8E 03 70 8E 05 70 A2
72A8- FE 8E 04 70 60 A5 4F 8D
72B0- 0C 70 A5 4E 8D 0B 70 20
72B8- DF 72 A5 4E CD 0B 70 F0
72C0- 10 A5 4F CD 0C 70 F0 09
72C8- C5 4E 90 02 A5 4E 29 03
72D0- 60 38 6A C9 7F 90 03 4C
72D8- B7 72 38 6A 4C B7 72 65
72E0- 4F C5 18 B0 0D 69 3F 65
72E8- 19 38 2A C5 4F B0 11 6D
72F0- 04 70 C9 BD B0 0E 4A C5
72F8- 19 90 09 6D 02 70 69 30
7300- 65 4F 38 6A 65 4E 85 4E
7308- A5 4F C5 4E B0 26 E9 23
7310- CD 02 70 B0 0A 38 6A E9
7318- AA E5 19 C5 4F B0 0C 65
7320- 4E 4A C9 7F 90 0E 38 6A
7328- ED 05 70 38 2A C9 60 B0
7330- 03 E5 18 2A 65 4E 85 4F
7338- 60 AD 00 C0 A2 00 8E 10
7340- C0 60
```

28685前，先鍵入POKE 28917，32：POKE 28918，115：POKE 28919，112：POKE 29001，32：POKE 29002，115：POKE 29003，112：CALL 28685便可。

在立體迷宮的程式中，其行110是輸入2個迷宮隨機亂數值，這可保證每次所產生的迷宮是完全不同。順帶一提是6502的架構是沒有隨機亂數的指令，那麽我們便要自行編寫一段隨機亂數產生程式，以放在＄4E和＄4F的2個8 BIT亂數值爲以後迷宮亂數產生的根本値（大家可參考電腦時代第一期〔眞假隨機數〕）。

若將行110改爲（POKE 78，110：POKE 79，50），則每次迷宮產生程序所得的迷宮圖案都會在CTRL—C（中止程序），及再執行程序時的迷宮圖案均是同一樣SEQUE—NCE。

比方，相信你們也會試過每次開動APPLE電腦後，鍵入PRINT RND (1)的指令，其所獲得的值在每次開機後均是完全相同之原理一樣。

此外，程式2行120的POKE 28710，96，其96爲6502機械碼的RTS（RETURN)指令，是控制副程式在迷宮產生後直接跳回BASIC控制系統下，而不致繼續執行老鼠仔尋路的機械碼副程式。

這亦說出，用者可在立體迷宮中，隨時以POKE 28710，32：CALL 28710去呼喚老鼠仔奔走於迷宮的副程式。

最後，本人所提供的立體迷宮亦是一個迷宮遊戲的雛模，而在迷宮中加入尋寶、尋出路、陷阱或甚至殺手等生色迷宮的概念，便有賴各位去發掘、思考和創作了。怎樣測試迷宮內的去路和障礙，那非要細心了解圖1的所示，可謂「FUN，FUN，FUN」呢！

編者按：讀者們留意，本文作者在編寫上列各程式的同時，還編寫了一個迷宮自動尋找出路的程式，程式本來不算長，只有8個SECTOR，但我們還是將它放入今期的程式磁碟中（FILE NAME：MAZE(AUTO RUN)），而且連其SO—URCE CODE也一併放入。

綫索

```
360  REM  ** LAYER 1 **
370  IF  PEEK (TT) = 255 THEN 510
380 A(0,1) = 1
390  IF  PEEK (TT + L) = 160 THEN
     A(0,0) = 1
400  IF  PEEK (TT + R) = 160 THEN
     A(0,2) = 1
410  REM  ** LAYER 2 **
420 TT = TT + F: IF  PEEK (TT) =
     255 THEN 510
430 A(1,1) = 1
440  IF  PEEK (TT + L) = 160 THEN
     A(1,0) = 1
450  IF  PEEK (TT + R) = 160 THEN
     A(1,2) = 1
460  REM  ** LAYER 3 **
470 TT = TT + F: IF  PEEK (TT) =
     255 THEN 510
480 A(2,1) = 1
490  IF  PEEK (TT + L) = 160 THEN
     A(2,0) = 1
500  IF  PEEK (TT + R) = 160 THEN
     A(2,2) = 1
510  REM  ** HGR MAZE **
520  HGR2
530  HCOLOR= 3: HPLOT 0,0 TO 279,
     0 TO 279,189 TO 0,189 TO 0,0
540  IF A(0,1) = 0 THEN 870
550  IF A(1,1) = 0 THEN 590
560  IF A(2,1) = 0 THEN 660
570  IF A(2,1) = 1 THEN 750
580  GOTO 180
590  REM  ** LEVEL 1 **
600  HPLOT 50,30 TO 229,30: HPLOT
     50,159 TO 229,159
610  IF A(0,0) = 1 THEN  GOSUB 97
     0: GOTO 630
620  HPLOT 0,0 TO 50,30 TO 50,159
      TO 0,189
630  IF A(0,2) = 1 THEN  GOSUB 99
     0: GOTO 880
640  HPLOT 279,0 TO 229,30 TO 229
     ,159 TO 279,189
650  GOTO 880
660  REM  ** LEVEL 2 **
670  HPLOT 100,69 TO 177,69 TO 17
     7,120 TO 100,120
680  IF A(1,0) = 1 THEN  GOSUB 10
     10: GOTO 710
690  IF A(0,0) = 1 THEN  GOSUB 10
     40: GOTO 710
700  HPLOT 0,0 TO 100,69 TO 100,1
     20 TO 0,189
710  IF A(1,2) = 1 THEN  GOSUB 10
     70: GOTO 880
720  IF A(0,2) = 1 THEN  GOSUB 11
     00: GOTO 880
730  HPLOT 279,0 TO 177,69 TO 177
     ,120 TO 279,189
740  GOTO 880
750  REM  ** LEVEL 3 **
760  HPLOT 131,89 TO 148,89: HPLOT
     131,100 TO 148,100
770  IF A(2,0) = 1 THEN  GOSUB 11
     30: IF A(0,0) = 0 THEN  GOSUB
     1250: GOTO 820
780  IF A(1,0) = 1 THEN  GOSUB 11
     60: GOTO 820
790  HPLOT 65,45 TO 110,75: HPLOT
     65,145 TO 110,115
800  IF A(0,0) = 1 THEN  GOSUB 12
     00: GOTO 820
810  HPLOT 0,0 TO 131,89 TO 131,1
     00 TO 0,189
820  IF A(2,2) = 1 THEN  GOSUB 12
     70: IF A(0,2) = 0 THEN  GOSUB
     1300: GOTO 880
```

```
830  IF A(1,2) = 1 THEN  GOSUB 13
     20: GOTO 880
840  HPLOT 214,45 TO 169,75: HPLOT
     214,145 TO 169,115
850  IF A(0,2) = 1 THEN  GOSUB 13
     60: GOTO 880
860  HPLOT 279,0 TO 148,89 TO 148
     ,100 TO 279,189: GOTO 880
870  HPLOT 0,0: CALL 62454
880  GOTO 180
890  REM  ** KEY INPUT **
900 K =  PEEK ( - 16384): IF K <
     128 THEN 900
910  POKE  - 16368,0: RETURN
920  REM  ** REVERSE DIRECION **
930  IF K = 149 THEN K = 136
940  IF K = 136 THEN K = 149
950  RETURN
960  REM  ** PLOT MAZE **
970  HPLOT 0,30 TO 50,30 TO 50,15
     9 TO 0,159
980  RETURN
990  HPLOT 279,30 TO 229,30 TO 22
     9,159 TO 279,159
1000  RETURN
1010  HPLOT 65,69 TO 100,69 TO 10
     0,120 TO 65,120
1020  HPLOT 0,0 TO 65,45 TO 65,14
     5 TO 0,189
1030  RETURN
1040  HPLOT 0,45 TO 65,45 TO 65,1
     45 TO 0,145
1050  HPLOT 65,45 TO 100,69 TO 10
     0,120 TO 65,145
1060  RETURN
1070  HPLOT 214,69 TO 177,69 TO 1
     77,120 TO 214,120
1080  HPLOT 279,0 TO 214,45 TO 21
     4,145 TO 279,189
1090  RETURN
1100  HPLOT 279,45 TO 214,45 TO 2
     14,145 TO 279,145
1110  HPLOT 214,45 TO 177,69 TO 1
     77,120 TO 214,145
1120  RETURN
1130  HPLOT 110,89 TO 131,89 TO 1
     31,100 TO 110,100
1140  HPLOT 65,45 TO 110,75 TO 11
     0,115 TO 65,145
1150  RETURN
1160  HPLOT 110,75 TO 131,89 TO 1
     31,100 TO 110,115
1170  HPLOT 65,75 TO 110,75 TO 11
     0,115 TO 65,115
1180  HPLOT 0,0 TO 65,45 TO 65,14
     5 TO 0,189
1190  RETURN
1200  HPLOT 0,45 TO 65,45 TO 65,1
     45 TO 0,145
1210  IF A(2,0) = 0 THEN  HPLOT 1
     10,75 TO 131,89 TO 131,100 TO
     110,115
1220  RETURN
1230  HPLOT 65,45 TO 131,89 TO 13
     1,100 TO 65,145
1240  RETURN
1250  HPLOT 0,0 TO 65,45: HPLOT 0
     ,189 TO 65,145
1260  RETURN
1270  HPLOT 169,89 TO 148,89 TO 1
     48,100 TO 169,100
1280  HPLOT 214,45 TO 169,75 TO 1
     69,115 TO 214,145
1290  RETURN
1300  HPLOT 279,0 TO 214,45: HPLOT
     279,189 TO 214,145
1310  RETURN
1320  HPLOT 169,75 TO 148,89 TO 1
     48,100 TO 169,115
1330  HPLOT 214,75 TO 169,75 TO 1
     69,115 TO 214,115
1340  HPLOT 279,0 TO 214,45 TO 21
     4,145 TO 279,189
1350  RETURN
1360  HPLOT 279,45 TO 214,45 TO 2
     14,145 TO 279,145
1370  IF A(2,2) = 0 THEN  HPLOT 1
     69,75 TO 148,89 TO 148,100 TO
     169,115
1380  RETURN  ■
```


訂閱收費表（包括郵費在內）

	本港	澳門、中國、東南亞區	其他地區	
雜誌加程式磁碟（十二期）	HK$475	HK$524	HK$554	平郵
	——	HK$654	HK$699	空郵
雜誌（十二期）	HK$120	HK$142	HK$164	平郵
	——	HK$242	HK$274	空郵

只需直接執行程式便可取得效果

使用更方便的BOOT DRIVE 2軟件

施　明

在6502電腦上運作的軟件數目不勝枚擧，可惜很多都以非標準的DOS來操作磁碟機，由於它們的磁碟操作系統不是標準系統格式的原故，在兩部磁碟機的環境下，若果要從任意一部磁碟機上取得心目中的DOS系統時，以手動方式來冷動第二部磁碟機是很困難的，爲了要不作複雜的硬件改動而達到理想目標，我提出一個別具心思的BOOT DRIVE 2程式。而這個程式可以容許用戶將標準的DOS置放於DRIVE 2上，在需要時按入＆PR＃6，D2來啓動DOS。

本程式包括兩個部份，第一個是以APPLESOFT編寫的載入程式，第二個是以組合語言編寫，它是一個包括四組建立指標，稽核句法命令等網路的程式。讀者在鍵入這兩個程式之後，只需將要置放在DRIVE 2上的標準DOS磁碟加以一個BYTE的修改即可永久地把它用作從DRIVE 2上啓動的DOS系統。

本程式由於考慮到將記憶體衝突問題降到最低把見，因此特意將程式放置到SLOT 0上RAM咭的第一個BANK。不過各位可以輕易地將它置放到其它位址上，而主記憶體上的＆向量指標（＄3F5－＄3F7）及進入點在＄9600上開始的處理程式（12個BYTE長）却需予以保留，但仍可以放到其它位址上去。

```
*********************
* DRIVE TWO BOOT *
*   A$2000, L$9B   *
*********************
2000- A9 01 8D 07 B7 A2 00 BD
2008- 8C D0 9D 00 96 E8 E0 0C
2010- D0 F5 A9 4C 8D F5 03 A9
2018- 00 8D F6 03 A9 96 8D F7
2020- 03 60 A2 04 BD 00 02 E0
2028- 02 F0 16 DD 00 02 D0 0E
2030- CA D0 F1 AD 05 02 C9 31
2038- F0 1A C9 32 F0 16 4C 06
2040- 96 C9 31 90 F9 C9 38 B0
2048- F5 69 90 85 FD 49 E0 85
2050- FF CA 90 D0 A0 FF 84 3E
2058- C8 84 3C 84 FC 84 FE A9
2060- 03 85 42 A5 FD 85 3D 85
2068- 3F A5 FF 85 43 20 2C FE
2070- A0 02 B9 98 D0 91 FE 88
2078- 10 F8 A0 39 AD 05 02 18
2080- 69 59 91 FE 6C FE 00 AF
2088- 8A 36 2C 44 2C 88 C0 4C
2090- 22 D0 2C 8A C0 4C C9 DE
2098- 2C 8A C0 29 C9
```

```
10  REM   ***********************
20  REM   **  INSTALL D2 BOOT  **
30  REM   **  INSTALL CODE FOR **
40  REM   **  BOOTING FROM D2  **
50  REM   **      INTO SLOT    **
60  REM   **      RAM  CARD    **
70  REM   ***********************
110  GOSUB 170
120  POKE  - 16247,0: POKE  - 162
     47,0
130  PRINT  CHR$ (4)"BLOAD DRIVE
     TWO BOOT,A$D000"
140  POKE 60,0: POKE 61,248: POKE
     62,255: POKE 63,255: POKE 66
     ,0: POKE 67,248
150  CALL 768
160  HOME :A$ = "BOOT2.DOS LOADED
     ": GOSUB 190: END
170  FOR AD = 768 TO 782: READ BY
     : POKE AD,BY: NEXT : RETURN
180  DATA   160,0,32,44,254,44,13
     6,192,32,0,208,44,138,192,96
190  HTAB 20 - ( LEN (A$)) / 2: PRINT
     A$: RETURN
```

鍵入程式須注意事項

大家從列表(1)中的主程式（機械碼）可見到，它是由＄D000開始而與ROM內置副程式的位址相衝突，因此在鍵入機械碼時需要以其它開始位址來進行鍵入，爲方便起見筆者建議由＄2000開始鍵入，完成後以BSAVE DRIVE TWO BOOT，A＄2000，L＄9B貯存回磁碟上去。基本上這個以組合語言編寫的程式功能是稽核BOOT DRIVE 2的命令句法"＆PR＃n，D2"是否正確，繼而將CONTROLLER咭上的ROM內置啓動程式移到＄2000位址開始作一些修改後便執行啓動DRIVE 2。

要製造在DRIVE 2上啓動的標準DOS磁碟，只要在一DRIVE 1上置放一只空磁碟，鍵入POKE 46855，2再按INIT INSTALL DRIVE 2 BOOT即可。POKE 46855是將DOS內的RWTS ＄B707改爲＄02，使不致在BOOT

使用更方便的BOOT DRIVE 2 軟件

DRIVE 2 時回轉到DRIVE 1 去。另方面本程式亦會將F8 ROM 上的內容移上RAM咭中。

要啓動DRIVE 2 BOOT時，只需鍵入RUN INSTALL DRIVE2 BOOT〔RET〕,隨後會顯示出訊息表示設置完成，在需要時只需按入＆PR＃n，D2，即可啓動插在任意SL－OT上的標準DRIVE 2 DOS，當然＆PR＃n，D1會啓動DRIVE 1。

最後要補充的是當我們的INSTALL 指標被覆蓋掉時（即$3F5－$3F7，$9600開始的12個位元組被破壞），而RAM 咭上的資料仍然保留的話，可以鍵入

```
CALL—151   〔RET〕
C088       〔RET〕
D000G      〔RET〕
3D0G       〔RET〕
```

使INSTALL 指標重新建立，不過若果在鍵入C088 時系統HANG UP的話，便代表RAM咭上的INSTALL 網路已被覆蓋掉，這時只有重新開機再執行了。

本文共包括有兩個程式列表，列表一爲機械碼主程式〔DRIVE TWO BOOT〕，列表二爲一個置放DRIVE TWO BOOT 主程式的輔助程式〔FILENAME 是INSTALL DRIVE 2 BOOT〕。

最後，筆者要提醒大家，這程式必須要連同16K RAM 咭使用。

編者按：我們分別在第十四、及十五期刊登了兩篇有關BOOT DRIVE 2的文章，有用改良硬件方法，亦有用軟件控制來達成。今期這篇文章，作者施君進一步利用軟件達成方法來隨意BOOT動DRIVE 2。施君所用的方法，比起梁卓傑君更爲方便，有效率。■

POKE 214,128 之破解方法

各位讀者可否曾經遇到一些用POKE 214，128 保護的程式呢？POKE 214，128 令到所有的APPLESOFT BASIC COMM－AND 都變成RUN指令，故無論你鍵入任何BASIC COMMAND，電腦都會RUN 在記憶體內的程式。

要破解此保護方法很簡單，因爲我發覺POKE 214，128 祇是使到APPLESOFT BASK COMMAND 變成RUN ，但DOS COMMAND卻能工作如常。因此，我們可從這方面入手！

其實，祇要我們用POKE 214，0，便能輕易將之破解！！但是如上所說，我們如果直接鍵入POKE 214，0，電腦一樣會將它當成RUN的。

但如果利用「程式一」去破解，無錯，我們是把POKE 214，128 破解了，但這時我們打入LIST，列出來的會是這個破解程式，故一樣沒效。（注意：程式一必須先SAVE入碟，用時用RUN ×××）爲何呢？因爲當我們RUN 這個程式時，會把受保護之程式洗去的。

```
10  REM  ******************
20  REM  *  FALSE BREAKER  *
30  REM  *  BY TONY CHANG  *
40  REM  ******************
50  HOME
60  POKE 214,0
70  END
```

程式 1

```
10  REM  ******************
20  REM  *    BREAKER      *
30  REM  *  BY TONY CHANG  *
40  REM  ******************
50  HOME
60  HTAB 5: VTAB 7: PRINT " TYPE
     THE PROTECTED FILE'S NAME :
     ";A$
70  INPUT A$
80  HOME
90  POKE 214,0
100  PRINT  CHR$ (4);"LOAD";A$;
110  END
```

程式 2

至於如何成功地破解POKE 214，128，又能完全地列出有保護的程式呢？程式二便能達到這目的了。

程式二首先要求用家輸入保護程式之FILE NAME，然後用POKE 214，0 破解POKE 214，128，最後再將保護程式LOAD出來。

但有一點要注意的是：使用本程式時必須將保護程式及此破解程式錄在同一隻碟上。

爲使大家更加明白，在此以程式三爲例說明之。

(1)首先將程式二及程式三錄在一隻磁碟上。

(2)鍵入LOAD TEST PROGRAM（程式三）

(3)再鍵入POKE 214，128

(4)此時除非你鍵入DOS COMMAND，否則電腦會將一切指令當爲RUN。（讀者可嘗試打入LIST／NEW等指令）

(5)RUN BREAKER

(6)應要求鍵入保護程式之FILE NAME──TEST PROGRAM

(7)待磁碟機轉動完畢，CURSOR出現後，鍵入LIST。

(8)成功了吧！你已經把保護程式列出來了。

```
10  REM  ** TEST PROGRAM **
20  HOME
30  HTAB 5: VTAB 5: PRINT "THIS I
     S A TEST PROGRAM "
40  HTAB 5: VTAB 6: PRINT "TYPE P
     OKE 214,128 AFTER SAVE THIS
     PROGRAM"
50  END
```

程式 3

■

專業室內設計師

可助你處理室內設計工作

任何房間、任何擺設，本系統皆適用

何守仁

你曾否嘗試改動自己的居住環境，如將睡房的傢俬移動一下，將它們重新組合？又或是於房中加插一些新傢俬？很麻煩？當搬動完畢後，發覺一些傢具不太脗合你原先的設計，又發覺牆脚不能再放下一件預先安排了的傢具，再搬動它們時，又沒有足夠空間應用，最後的結果，相信大家不用說也知道是搞得亂七八糟，一頭烟了，沒有地方再擺下一件東西，甚至出入口的大門也被傢具阻著，不能進出，此時，我們只能說一句：「噢，我的天啊！」上帝也不能打救你。解決這個問題,當然有更好的方法,將各傢具依實物呎吋縮小畫在平面上設計？再重新安排？這個方法當然比上面所述的方法好，但這樣你難免在設計過程中發生錯誤，將比例用錯了，這是最常見的錯誤，或許你沒用錯比例，但畫了一幅設計圖後不滿，當然要畫第二幅，第三幅，最後又發覺第一幅所畫的好些，如此種種的方法，費時失事，都不是明智的抉擇。你們何不試用筆者設個專業室內設計師PROFESSIONAL ARCHITECT 程式，讓它分担你這項工作，讓它來作你的設計師呢？PROFESSIONAL ARCHITECT 可以助你解決困難，它具有先進的處理程式和設計概念，能夠有效地重整或重新設計一幅平面，你可利用它為你設計理想家居，而且它有除錯功能，易於修改，並可以於設計每所房間時處理多達50件傢具，包括處理門窗位置，準確度以吋為單位。

PROFESSIONAL ARCHITECT 有三大項主要功能：

(1)圖像編輯
(2)磁碟檔案管理
(3)印出圖像（適合ESPON或相融系列印字機）

當程式運行時，螢光幕會顯示FIG.1的圖像，它指示程式的各功能名稱及命令碼。若此時你想開始使用程式的巨大威力，便要按"N"鍵或"O"鍵。

開始設計

N——創造一所新房間

當選擇N鍵後，螢光幕會顯示讀取資料的訊息，程式會要求輸入資料，如此便可決定房間的大小呎吋，長濶，每平方格的面積（THE SPACING BETWEEN GRID LINES）和房間的名稱，如厨房、睡房等。但決定的呎吋絕不可超過99呎，否則兩個"**"符號將會出現螢光幕上，拒絕接納輸入的資料。因程式要求的長濶範圍是大於0呎及小於99呎。

O——從磁碟取回舊有房間的設計圖

```
FUNCTIONS ASSIGNED TO THE KEYBOARD:

A-ALTER        K-RIGHT        U-UNLATCH
B-BOX          L-LATCH        V-VERIFY
C-COPY         M-DOWN         W-WINDOW
D-DOOR         N-NEW          X-EXIT W/O DOOR
E-ERASE        O-OLD          Y-YES
F-FIX          P-PLACE        *-CIRCLE
G-GET DATA     Q-QUIT         #-GRID & TITLE
H-HARDCOPY     R-REDRAW       <-SMALLER STEP
I-UP           S-SAVE         >-LARGER STEP
J-LEFT         T-TURN         ?-HELP
```

圖㈠，本程式的各項控制指令一覽表

當選擇O鍵後，螢光幕會出現一幅有關磁碟檔案處理的畫面，用GET指令選擇適當的設計圖則，如果選擇錯誤，如選擇一個不存在的設計圖，又或是用錯SAVE和REVISE 指令，將會收到錯誤的訊息。經適當的選擇後，設計圖的編號和名稱會以閃爍字元出現，直至GET指令找到正確的設計圖，才以反正形式印出。此時可以使用SAVE和REVISE指令了，按下SPACE BAR，設計圖像會顯示螢光幕上，若再按下SPACE BAR，它的功能便等於NEW指令，將設計圖清除。

PROFESSIONAL ARCHITECT 程式除了上述兩個功能鍵外，還有許多其他的功能，讓筆者將它們分門別類向各位一一闡述。

圖像編輯

在圖像編輯過程中，螢幕的左下方會出現"EDIT"的字樣，並處於高解像模式中，若選用某個編輯功能，則"EDIT"字樣將被該功能的名稱代替，直至完成該項功能的編輯過程。

游標控制功能

程式使用兩個不同的游標，一個為移動游標，用符號（×）代替，另一個為固定游標，用符號（+）代替。固定游標是用來作參考，給移動游標一個固定的參考點，於移動時有所依據（於程式開始時，移動游標與固定游標均是設於房間的一角）。移動游標每次可移動1吋、2吋或3吋，但絕不可超出房間決定的範圍。移動距離可用＜及＞鍵來調查。

控制游標，可利用I，J，K，M鍵達成，而F鍵是用來專負責固定游標的移動。按下G鍵，會顯示游標移動的距離

。以下是各游標控制功能鍵的用途：

- I　將移動游標向上移一步
- J　將移動游標向右移一步
- K　將移動游標向左移一步
- L　將移動游標向下移一步
- ＜　將移動游標的移動距離減1吋
- ＞　將移動游標的移動距離加1吋
- F　將固定游標（十）移至現時移動游標的位置上
- G　將固定游標與移動游標的距離顯示出來

注意：當按G鍵時，固定游標與移動游標的水平距離、垂直距離和對角距離均會被顯示，而I，J，K，M，＜和＞指令仍然有效。當移動游標超出房間範圍，會有閃動訊息顯示。此鍵非常有用，大家宜加留意。

物件創造功能

只要利用固定游標（十）與移動游標（×），和下列所述的功能鍵，便可以創造自己的傢具形狀，如窗、門、椅等。當創造一件傢具完畢後，我們需要另外一組的功能鍵將其移至正確的位置。而傢具的大小呎吋，是根據兩個游標的移動距離決定，形狀可利用B，＊，D，W和× 功能鍵決定。若固定游標與移動游標互相重叠，則沒有物件被創造，例如已創造完畢最後一件傢具。方形與圓形的物體只可以固定游標於房間位置之內時繪製或創造，而窗和門的繪製，固定游標必須位於內外牆之間。現在看看功能鍵的用途：

P——繪製一個方形；固定游標與移動游標需於方形的對角位置。

＊——繪製一個圓形；固定游標為圓形的中心點，而移動游標與固定游標距離為該圓形之半徑值。

D——繪製一隻門；固定游標位於門旋轉軸，而移動游標與其距離為門的濶度。

W——繪製一隻窗；固定游標與移動游標距離為該窗之濶度。

×——繪製一個出入口，功能如D一樣，但不會繪出門。

修改及移動功能

每次使用修改及移動功能之前，必先設定被移動或修改的物體是什麼？物體被移動前需要先給它一個標記（LAT－CHED），移動完畢後需取消該標記（UNLATCHED）。而A、C、E、P、T、U及V 功能鍵只會對被加上標記的物體工作，若該物體未加標記，則上述各功能鍵將不會對該物體有所行動。若固定游標位於房間之內，則房間內的方形物體和圓形物體才可接受指令控制，若固定游標位於外牆與內牆之間，則指令可將窗、門移動與修改，此時指令不可控制房內的物件。相對固定游標位於房內時亦然。

看看各功能鍵的用途：

L——為物件加上標記；當移動游標於某物件的包圍下使用L指令，可使該物件加上標記，並有一個方形符號出現於該物件的中央。

U——為所有加上標記的物件除去標記；所有方形符號將會被刪掉。

A——改變被標記物件的形狀；受改變的物件會有一個十符號出現在該物件的標記位置，此時可以使用I，J，K，M，＜，＞，F和G 功能鍵，將物件的形狀大小修改。若按A鍵，則取消該物件，改動符號消失，完成改動工作。當改動完一件物件後，按下A鍵會自動跳至另一個需要接受改動的物件，繼續改動工作，但此時固定游標與移動游標將會重叠。

C——抄錄已標記之物件；如此你便可抄錄多幾份相同之物體，免了重新繪畫。所有已標記之物件俱可使用C指令，而新抄錄的物件可利用固定游標和移動游標至新的位置，但移動門與窗時只可於內外牆之間移動。

E——取消已標記之物件；你可以利用E指令將錯誤之物件取消。

P——移動已標記之物件；所有被標記之物件均可利用固定游標與移動游標移至新的位置。甚至可將物件移出房外，此時其物件之標記會位於貼近外牆的位置，表示該物件之存在。而該物件之位置是可以恢復或再次移動至新位置的。

T——轉動已標記之物件；這個指令只對方形的物體與圓形的物體有影響，如果固定游標與移動游標重叠，則被標記之物件以反時鐘方向轉動90°，以游標為轉動之中心點。若固定游標與移動游標不重叠，則物件以其中心點轉動，直至與兩個游標所成的直線平行。

V——顯示各被標記物件的大小呎吋；這個命令將房間及多至5項已標記之大小呎吋以表列式顯示出來。

附加功能鍵：

R——重設房間的設計與傢具；當修改或重定位置，不同之物件，會令部份之畫面消失。R指令可令你重新繪製設計圖，它的功能是清除房間內未被標記之物件，或未被完全擦除之物件及任何超出房間範圍外之物件。

#——改動虛線之間格距離及設計圖名稱；#指令可以修改或刪除虛線（GRID LINE）和設計圖名稱。改動之設計圖名稱是在圖的底部，若刪除名稱，則螢幕左端的提示（LOGO）亦會消失。任何改動，都會重繪設計圖，並進入圖像編輯功能。若被改動之虛線間隔等於0，則此時程式將不進入圖像編輯功能。

?——顯示各命令的用處；它可以於編輯功能中顯示不同編輯指令的用處。

磁碟檔案管理

S指令是用來進入磁碟檔案管理程式部份，我們可利用

它改動已有的設計圖，並有觀看和取消功能。在一隻空磁碟中，PROFESSIONAL ARCHITECT 程式最多可處理42幅設計圖或房間。

當使用S鍵後，螢光幕顯示檔案管理畫面。其中的DE—LETE指令和GET 指令需要輸入設計圖編號，而SAVE指令則需要設計圖之名稱。REVISE指令是用來觀看磁碟內已有之設計圖，選擇之設計圖編號與名稱將以反白印出（上述的GET指令與SAVE 指令亦是一樣）。使用SAVE會使新的設計圖蓋掩舊有的設計圖，若設計圖名稱相同。而若取消一幅設計圖，檔案管理內部會自動將往後設計圖的編號移前，使得被取消之設計圖沒佔有檔案編號。

NEW指令容許你重新設計一所房間，QUIT指令則脫離檔案管理狀態。以上所有之命令，都會有所提示，得到許可後才工作，所以不用驚怕按錯鍵盤碼。

列印功能

使用H指令可以令你在高解像的畫面由一部MX—80 印字機列印出來，相融之機種亦可。

H指令可使你的印列一幅 3 吋× 4 吋或 6 吋× 8 吋的圖像。你亦可用R指令將要被列印之畫面重畫。而在列印的過程中，SPACE BAR鍵是用來停止列印。

若你所擁有的印字機不是MX— 80一樣的機種，你亦可將圖像先儲存入磁碟，再將圖像用其他牌子的印字機印列出來。

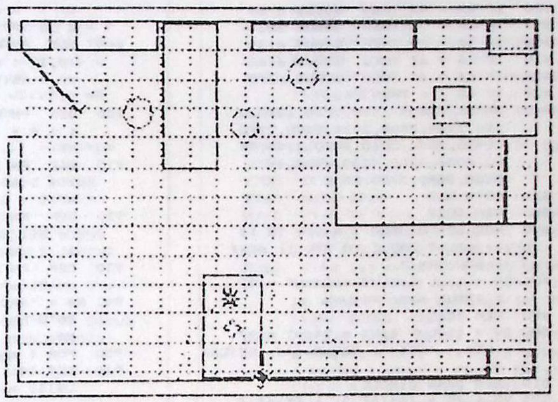
跳出功能

Q指令可令你自由跳出PROFESSIONAL ARCHIT—ECT，在圖像編輯或磁碟檔案管理中，可利用Q指令跳出，而每次跳出前，電腦會首先發問你是否作跳出動作。

輸入程式

PROFESSIONAL ARCHITECT 程式共分三部份：一個APPLESOFT 程式，一個包含MENU 的文字檔和一個包含游標形狀的高解像SHARP TABLE。因PROFESS—IONAL ARCHITECT 程式需要在磁碟內進行輸入輸出的工作，而且需要磁碟有一定的空位，故筆者建議使用一隻只載有PROFESSIONAL ARCHITECT程式的磁碟。留意本程式不適合於PRODOS 使用。

本圖是電腦時代讀者服務部的平面設計圖，經常到來服務部的讀者應當感到熟識。如果閣下認爲有更好的擺設方式，不妨給與服務部負責人一些意見，他們必定多謝你。



OFFICE

各位於輸入程式時，只需先打入LISTING 1的APPLE—SOFT程式，以SAVE PROFESSIONAL ARCHITECT將程式載入磁碟，然後再打入LISTING 2 程式，並以SAVE MENU. CREATE載入磁碟，之後你需要以RUN 指令執行此程式，獲得一個以ARCHITECT. MENU 爲名稱的文字檔，最後將LISTING 3 程式打入，並以SAVE ALPH. CR—EATE 載入磁碟，再以RUN 執行程式，獲得一個名爲ARCHITECT. ALPH的SHARP TABLE檔案。

此時，你將不再需要MENU. CREATE與ALPH. CRE—ATE檔案，你需要的只是PROFESSIONAL ARCHIT—ECT，ARCHITECT. ALPH和ARCHITECT. MENU，只要這三個檔案在同一磁碟上，你便可執行PROFESSION—AL ARCHITECT如常了。■

LISTING #1

```
10  REM  ********3.12.85*********

20  REM  *PROFESSIONAL ARCHITECT*

30  REM  *    BY COMPUHOUSE     *

40  REM  *  COPYRIGHT (C) 1985  *

50  REM  *  BY COMPUTING AGE    *

60  REM  *                      *

70  REM  ************************

80  HOME : VTAB 6: HTAB 8: PRINT
     "PROFESSIONAL ARCHITECT": IF
     PEEK (104) <  > 64 THEN  POKE
     103,1: POKE 104,64: POKE 163
     84,0: PRINT  CHR$ (4)"RUN PR
     OFESSIONAL ARCHITECT"
90  POKE 105,0: POKE 106,8: POKE
     107,0: POKE 108,8: POKE 109,
     0: POKE 110,8
100 ST$ = "T":SA = 8192:BY = 0:TR
     = 0:OC = 0:FL = 0:N1 = 1024
    :N1% = 255:N2% = 128:N3% = 3
    2:N2 = 8063:N3 = 984:E$ =  CHR$
    (27)
110 ST = 2:MV = 0:FH = 0:FV = 0:A
    S = .86:HS = 3:VS = 3:C = 0:
    TF = 0:TI = 0
120 C$ = "Z":XC = 0:YC = 0:XF = 0
    :YF = 0:XS = 0:YS = 0:L = 0:
    N = 0:CL = 3:CR = 0:F = 0:G =
    0:P5 = .5:Q$ =  CHR$ (34):PR
     = .017453:P4 = .7854:A = 0:
    WS = 6
130  GOTO 280
140  REM MOVING CURSOR
150  XDRAW 3 AT HS + HR * XC,VS +
    VR * YC: RETURN
160  REM FIXED CURSOR
170  XDRAW 2 AT HS + HR * XF,VS +
    VR * YF: RETURN
180  REM MOVES
190 CA =  PEEK ( - 16384) - 128: IF
    CA < 0 THEN 190
200  POKE  - 16368,0
210  IF CA = 73 AND YC > Y1 THEN
     GOSUB 140:YC = YC - ST:FL =
    1:FV = 0: RETURN
220  IF CA = 74 AND XC > X1 THEN
     GOSUB 140:XC = XC - ST:FL =
    1:FH = 0: RETURN
230  IF CA = 75 AND XC < X2 THEN
     GOSUB 140:XC = XC + ST:FL =
    1:FH = 0: RETURN
240  IF CA = 77 AND YC < Y2 THEN
     GOSUB 140:YC = YC + ST:FL =
    1:FV = 0: RETURN
250  IF CA = 74 OR CA = 75 THEN F
    H = 1
260  IF CA = 73 OR CA = 77 THEN F
    V = 1
270 FL = 0: RETURN
280  REM PREAMBLE
290  GOSUB 770
300  GET C$
310  IF C$ = "N" THEN  GOSUB 3190

320  IF C$ = "O" THEN  GOSUB 4480
```

```
330  IF N = 0 THEN 300
340  REM MONITOR
350  GOSUB 180
360  IF FL = 1 THEN MV = 1: GOSUB
      140: GOTO 350
370 CS = CA - 64
380  IF CA = 42 THEN  GOSUB 5750
390  IF CA = 35 THEN  GOSUB 5830
400  IF CA = 60 THEN  GOSUB 6130
410  IF CA = 62 THEN  GOSUB 6160
420  IF CA = 63 THEN  GOSUB 6190
430  IF CS < 0 THEN 350
440  ON CS GOSUB 1500,1730,1890,2
     050,2260,2400,2420,2690,2990
     ,2990,2990,3010,2990,3190,44
     80,4050,4200,4240,4480,4920,
     5200,5280,5450,5530
450  GOTO 350
460  REM BUSY
470  HCOLOR= 0: FOR I = 184 TO 19
     1: HPLOT 252,I TO 279,I: NEXT
      : HCOLOR= 3
480 IV = 1:VL = 0:VT = 24:HT = 37
     : GOSUB 640: RETURN
490  REM EDIT
500 IV = 1:VL = 0:VT = 24:HT = 37
     :ST$ = "EDIT": GOSUB 640: RETURN

510  REM DRAW CIRCLE
520 XG = P5 * (A%(AI,2) + A%(AI,4
     )):YG = P5 * (A%(AI,3) + A%(
     AI,5))
530 D = P5 * (A%(AI,4) - A%(AI,2)
     ):D% = VR * D:WA = 0:WB = 2 *
     WS:AG = 15 * PR
540  IF D% > 10 THEN AG = 12 * PR

550  IF D% > 25 THEN AG = 9 * PR
560  IF D% > 35 THEN AG = 8 * PR
570  IF D% > 45 THEN AG = 6 * PR
580  IF D% > 55 THEN AG = 5 * PR
590  FOR I = T1 TO T2 STEP AG
600 X = (XG + D *  COS (I)):Y = (
     YG - D *  SIN (I))
610  IF X < WA OR Y < WA OR X > W
     B + L1 OR Y > WB + W1 THEN 6
     30
620  HPLOT HS + HR * X,VS + VR *
     Y
630  NEXT : RETURN
640  REM ALPHABET
650  IF VL = 0 THEN BA = N2 + N2%
      * VT - N3 *  INT ((VT - 1) /
      8) + HT
660  FOR I = 1 TO  LEN (ST$): IF
     VL = 1 THEN BA = N2 + N2% *
     VT - N3 *  INT ((VT - 1) / 8
     ) + HT
670 CH =  ASC ( MID$ (ST$,I,1)) -
      N3%
680  IF CH = 0 THEN 730
690  IF IV = 1 THEN 720
700  FOR J = 0 TO 7: POKE BA + N1
      * J,AL%(CH,J): NEXT
710  GOTO 730
720  FOR J = 0 TO 7: POKE BA + N1
      * J,N1% - AL%(CH,J): NEXT
730 BA = BA + (VL = 0):VT = VT +
      (VL = 1): NEXT : RETURN
740  REM MENU
750  PRINT L$(6)L$(7)L$(8)L$(9)L$
     (10)L$(11)L$(12)L$(13)L$(14)
     L$(15)
760  RETURN
770  REM INITIALIZE
780  VTAB 12: PRINT  SPC( 39);: HTAB
      13: FLASH : PRINT " READING
      MENU ": NORMAL
790 D$ =  CHR$ (4): REM CTRL-D
800 CR$ =  CHR$ (13): REM C/R
810 AD = 0:LN = 1179
820 AD =  PEEK (107) + 256 *  PEEK
      (108)
830  PRINT D$"BLOAD ARCHITECT.ALP
     H,A"AD
840  POKE 110, INT ((AD + LN) / 2
     56)
850  POKE 109,AD + LN - 256 *  INT
      ((AD + LN) / 256)
860 NL = 67: DIM A%(52,7),L$(NL),
     D(5),F$(5),E(1),G$(3),V%(383
     )
870  DEF  FN IN(W) = XF > = WS AND
      XF <  = WS + L1 AND YF >  =
     WS AND YF <  = WS + W1
880  DEF  FN OT(X) = XF <  = WS OR
      XF >  = WS + L1 OR YF <  = W
     S OR YF >  = WS + W1
890  DEF  FN HO(U) = A%(U,5) - A%
     (U,3) < WS + .1: DEF  FN VE(
     V) = A%(V,4) - A%(V,2) < WS +
     .1
900  DEF  FN BB(B) = MX * (B > MX
     ) + B * (B >  = 0 AND B <  =
     MX)
910  DEF  FN BD(D) = MY * (D > MY
     ) + D * (D >  = 0 AND D <  =
     MY)
920  DEF  FN VH(J) = A%(I,J) < WS
      - ST: DEF  FN RH(J) = A%(I,
     J) > WS + L1 + ST
930  DEF  FN BV(J) = A%(I,J) > WS
      + W1 + ST
940 M$ = "ARCHITECT.MENU": PRINT
     D$"OPEN "M$: PRINT D$"READ "
     M$
950  FOR I = 0 TO 1:L$(I) = ""
960  GET C$: IF C$ <  > CR$ THEN
     L$(I) = L$(I) + C$: GOTO 960

970 X =  FRE (0): NEXT
980  FOR I = 2 TO NL: INPUT L$(I)
     : NEXT
990  FOR H = 832 TO 880: INPUT X:
      POKE H,X: NEXT
1000  FOR I = 1 TO 2: INPUT X,Y: POKE
     X,Y: NEXT
1010  FOR J = 768 TO 831: INPUT X
     : POKE J,X: NEXT
1020  PRINT D$"CLOSE "M$: SCALE=
     1
1030  ONERR  GOTO 1110
1040  DIM DF$(42):M$ = "ARCHITECT
     .DIR": PRINT D$"OPEN "M$: PRINT
     D$"READ "M$
1050  INPUT DF$(0):DF =  VAL (DF$
     (0)): IF DF = 0 THEN 1070
1060  FOR I = 1 TO DF: INPUT DF$(
     I): NEXT
1070  PRINT D$"CLOSE "M$
1080  VTAB 1: PRINT L$(3)"*" SPC(
     38)"*"L$(4)"*" SPC( 38)"*"L$
     (5)"*" SPC( 38)"*"L$(3)
1090  VTAB 8: CALL  - 958: PRINT
     : PRINT L$(0): PRINT : GOSUB
     740: PRINT : PRINT L$(16);
1100  RETURN
1110  CALL  - 3288: POKE 216,0: PRINT
     D$"CLOSE":DF = 0: GOSUB 4810
     : GOTO 1080
1120  REM INPUT ANYTHING
1130  CALL 54572: FOR H = 512 TO
     768: IF  PEEK (H) <  > 0 THEN
     1150
1140 IN$ = "":AD =  VAL (IN$) +  PEEK
      (131) + 256 *  PEEK (132): POKE
     AD,H - 512: POKE AD + 1,0: POKE
     AD + 2,2:IN$ =  MID$ (IN$,1)
     :H = 768
1150  NEXT : RETURN
1160  REM TEXT
1170  HOME : POKE  - 16303,0: RETURN

1180  REM GRAPHICS
1190  POKE  - 16304,0: RETURN
1200  REM FIXED & MOVING CURSORS
1210  GOSUB 140: GOSUB 160: RETURN

1220  REM RECTANGLE
1230 MX = A%(2,4):MY = A%(2,5)
1240 XL = A%(AI,2):YT = A%(AI,3):
     XR = A%(AI,4):YB = A%(AI,5)
1250  IF A%(AI,7) <  > 0 AND A%(A
     I,0) = 2 THEN  GOSUB 5080: RETURN

1260 XL =  FN BB(XL):XR =  FN BB(
     XR):YT =  FN BD(YT):YB =  FN
     BD(YB)
1270 LX = HS + HR * XL:RX = HS +
     HR * XR:TY = VS + VR * YT:BY
      = VS + VR * YB
1280  IF A%(AI,0) = 3 THEN  GOSUB
     5490: RETURN
1290  IF A%(AI,0) = 5 THEN T1 = 0
     :T2 = 359 * PR: GOSUB 510: RETURN

1300  HPLOT LX,TY TO RX,TY TO RX,
     BY TO LX,BY TO LX,TY
1310  IF A%(AI,0) = 4 THEN  GOSUB
     5550
1320  RETURN
1330  REM GRID SPACING
1340 XM = 262:YM = 172
1350 HR = XM / (L1 + 2 * WS):VR =
     YM / (W1 + 2 * WS)
1360  IF VR < AS * HR THEN HR = V
     R / AS: GOTO 1380
1370 VR = AS * HR
1380 HD = S1 * HR:VD = S1 * VR:XZ
      = WS * HR:YZ = WS * VR
1390 XM = HR * (L1 + WS):YM = VR *
     (W1 + WS)
1400 X1 = WS - 1:X2 = L1 + WS + 1
     :Y1 = WS - 1:Y2 = W1 + WS +
     1: RETURN
1410  REM GRID LAYOUT
1420  IF YZ + VD > YM - 1 THEN 14
     60
1430  FOR I = VS + YZ + VD TO VS +
     YM - 1 STEP VD
1440  FOR J = HS + XZ + HS TO HS +
     XM - 1 STEP HS
1450  HPLOT J,I: NEXT : NEXT
1460  IF XZ + HD > XM - 1 THEN  RETURN

1470  FOR I = HS + XZ + HD TO HS +
     XM - 1 STEP HD
1480  FOR J = VS + YZ + VS TO VS +
     YM - 1 STEP VS
1490  HPLOT I,J: NEXT : NEXT : RETURN

1500  REM A...ALTER
1510  IF L = 0 OR A%(0,1) = 2 THEN
      RETURN
1520 A = 1:ST$ = "A": GOSUB 460: FOR
     AI = 3 TO A%(0,1):OB = A%(AI
     ,0)
1530  IF A%(AI,1) = 0 THEN 1720
1540  GOSUB 3070
1550  GOSUB 180
1560  IF FL = 1 THEN MV = 1: GOSUB
     140: GOTO 1550
1570  IF CA = 65 THEN 1630
1580  IF CA = 71 THEN  GOSUB 2420

1590  IF CA = 70 THEN  GOSUB 2400

1600  IF CA = 60 THEN  GOSUB 6130

1610  IF CA = 62 THEN  GOSUB 6160

1620  GOTO 1550
1630  IF MV = 0 THEN  GOSUB 3070:
     GOTO 1720
1640  IF  FN OT(XF) = 0 AND (OB =
     3 OR OB = 4) THEN 1550
1650 A = 2: GOSUB 1200: HCOLOR= 0
     :CR = 3:CL = 0: GOSUB 4330: HCOLOR=
     3:CR = 0:CL = 3
1660  IF OB = 2 THEN  GOSUB 1730
1670  IF OB = 3 THEN  GOSUB 5450
1680  IF OB = 4 AND A%(AI,6) = 0 AND
     A%(AI,7) = 0 THEN  GOSUB 553
     0: GOTO 1710
1690  IF OB = 4 THEN  GOSUB 2050
1700  IF OB = 5 THEN  GOSUB 5750
1710 A = 0: GOSUB 3070:A = 1
1720  NEXT :A = 0: GOSUB 490: RETURN

1730  REM B...BOX
1740  IF MV = 0 THEN  RETURN
1750  IF A = 0 THEN ST$ = "B": GOSUB
     1820:A%(AI,0) = 2
1760 A%(AI,2) = XC:A%(AI,4) = XF
1770  IF XC > XF THEN A%(AI,2) =
     XF:A%(AI,4) = XC
1780 A%(AI,3) = YC:A%(AI,5) = YF
1790  IF YC > YF THEN A%(AI,3) =
     YF:A%(AI,5) = YC
1800  IF A = 0 THEN A%(AI,6) = 1:
     A%(AI,7) = 0
```

專業室內設計師

```
1810  GOSUB 1220: GOSUB 1860: RETURN

1820  REM ADD
1830  IF C = 0 THEN  GOSUB 460: GOSUB
   1200
1840 AI = A%(0,1) + 1: IF AI > 52
    THEN  GOSUB 1160: VTAB 12: HTAB
   10: FLASH : PRINT " "L$(42);
   : NORMAL :DL = 3000: GOSUB 2
   380: GOSUB 1180: GOSUB 490:C
    = 0: GOSUB 1200: POP : RETURN

1850 A%(0,1) = AI:A%(AI,1) = 0: RETURN

1860  REM RETURN
1870 MV = 0: IF A = 0 THEN  GOSUB
   490
1880  GOSUB 1200: RETURN
1890  REM C...COPY
1900  IF L = 0 OR MV = 0 OR A%(0,
   1) = 2 THEN  RETURN
1910 C = 1:ST$ = "C": GOSUB 460: GOSUB
   1200:XH = XC - XF:YH = YC -
   YF
1920  FOR AN = 3 TO A%(0,1):OB =
   A%(AN,0)
1930  IF A%(AN,1) = 0 THEN 2040
1940  IF (OB = 3 OR OB = 4) AND  FN
   OT(XF) = 0 THEN 2040
1950  IF (OB = 2 OR OB = 5) AND  FN
   IN(XF) = 0 THEN 2040
1960 XE = XH:YE = YH
1970  IF (OB = 3 OR OB = 4) AND  FN
   HO(AN) THEN YE = 0: IF YF >
   WS AND YF < WS + W1 THEN 204
   0
1980  IF (OB = 3 OR OB = 4) AND  FN
   VE(AN) THEN XE = 0: IF XF >
   WS AND XF < WS + L1 THEN 204
   0
1990  GOSUB 1820:A%(AI,0) = A%(AN
   ,0)
2000 A%(AI,2) = A%(AN,2) + XE:A%(
   AI,3) = A%(AN,3) + YE
2010 A%(AI,4) = A%(AN,4) + XE:A%(
   AI,5) = A%(AN,5) + YE
2020 A%(AI,6) = A%(AN,6):A%(AI,7)
    = A%(AN,7)
2030  GOSUB 4330
2040  NEXT :C = 0: GOSUB 1860: RETURN

2050  REM D...DOOR
2060  IF  FN OT(XF) = 0 OR MV = 0
   THEN  RETURN
2070 ST$ = "D": IF F = 1 THEN ST$
    = "X"
2080  IF A = 0 THEN  GOSUB 1820:A
   %(AI,0) = 4
2090 A6 = 0:A7 = 0: GOSUB 2120
2100  IF F = 1 THEN A%(AI,6) = 0:
   A%(AI,7) = 0
2110  GOSUB 1220: GOSUB 1860: RETURN

2120  REM POSITION
2130  IF XF <  = WS AND YF < YC THEN
   A2 = 0:A4 = WS:A3 = YF:A5 =
   YC:A7 = 1
2140  IF XF >  = WS + L1 AND YF <
   YC THEN A2 = WS + L1:A4 = 2 *
   WS + L1:A3 = YF:A5 = YC:A7 =
   2
2150  IF XF <  = WS AND YF > YC THEN
   A2 = 0:A4 = WS:A3 = YC:A5 =
   YF:A7 = 4
2160  IF XF >  = WS + L1 AND YF >
   YC THEN A2 = WS + L1:A4 = 2 *
   WS + L1:A3 = YC:A5 = YF:A7 =
   3
2170  IF YF <  = WS AND XF < XC THEN
   A2 = XF:A4 = XC:A3 = 0:A5 =
   WS:A6 = 1
2180  IF YF >  = WS + W1 AND XF <
   XC THEN A2 = XF:A4 = XC:A3 =
   WS + W1:A5 = 2 * WS + W1:A6 =
   4
2190  IF YF <  = WS AND XF > XC THEN
   A2 = XC:A4 = XF:A3 = 0:A5 =
   WS:A6 = 2
2200  IF YF >  = WS + W1 AND XF >
   XC THEN A2 = XC:A4 = XF:A3 =
   WS + W1:A5 = 2 * WS + W1:A6 =
   3
2210 A%(AI,2) = A2:A%(AI,3) = A3:
   A%(AI,4) = A4:A%(AI,5) = A5:
   A%(AI,6) = A6:A%(AI,7) = A7:
    RETURN
2220  REM ARC
2230 CQ = CI * 2 - 1
2240  HPLOT HS + XG * HR,VS + YG *
   VR TO HS + (XG + D *  COS (C
   Q * P4)) * HR,VS + (YG + D *
    SIN (CQ * P4)) * VR
2250 AG = 10 * PR:WA = WS:WB = WS
   : GOSUB 590: RETURN
2260  REM E...ERASE
2270  IF L = 0 OR A%(0,1) = 2 THEN
    RETURN
2280 ST$ = "E": GOSUB 460: GOSUB
   1200: HCOLOR= 0:CR = 3:CL =
   0:E = 1: GOSUB 2320
2290 DL = 1000: GOSUB 2380: GOSUB
   1160: VTAB 12: HTAB 13: PRINT
   L$(34);
2300  GET C$: IF C$ = "Y" THEN E =
   0:L = 0
2310  HCOLOR= 3:CR = 0:CL = 3: GOSUB
   1180: GOSUB 2320: GOSUB 490:
    GOSUB 1200: RETURN
2320  REM ERASING
2330  FOR AI = 3 TO A%(0,1)
2340  IF A%(AI,1) = 0 THEN 2370
2350 A%(AI,0) = E * A%(AI,0):A%(A
   I,1) = E * A%(AI,1)
2360  IF E = 1 THEN  GOSUB 4330
2370  NEXT : RETURN
2380  REM DELAY
2390  FOR I = 1 TO DL: NEXT : RETURN

2400  REM F...FIX
2410  GOSUB 160:XF = XC:YF = YC: GOSUB
   160:MV = 0: RETURN
2420  REM G...GET DATA
2430  GOSUB 1160
2440  INVERSE : HTAB 5: PRINT L$(
   19): NORMAL : VTAB 4: PRINT
   L$(1): PRINT L$(2): VTAB 9: PRINT
   L$(17)
2450  INVERSE : VTAB 13: HTAB 9: PRINT
   "V";: HTAB 15: PRINT L$(37)
2460  HTAB 9: PRINT "E";: HTAB 12
   : PRINT "D": HTAB 9: PRINT "
   R";: HTAB 13: PRINT "I"
2470  HTAB 9: PRINT "T";: HTAB 14
   : PRINT "A": HTAB 9: PRINT "
   I";: HTAB 15: PRINT "G"
2480  HTAB 9: PRINT "C";: HTAB 16
   : PRINT "O": HTAB 9: PRINT "
   A";: HTAB 17: PRINT "N"
2490  HTAB 9: PRINT "L";: HTAB 18
   : PRINT "A": HTAB 19: PRINT
   "L": NORMAL
2500  GOSUB 2570
2510  GOSUB 180: GOSUB 2570
2520  IF FL = 1 THEN MV = 1: GOSUB
   140: GOTO 2510
2530  IF CA = 32 THEN  GOSUB 1180
   : RETURN
2540  IF CA = 60 THEN  GOSUB 6130

2550  IF CA = 62 THEN  GOSUB 6160

2560  GOTO 2510
2570  REM DISTANCES
2580 XD = XC - XF:YD = YC - YF
2590  IF FH = 1 THEN  FLASH
2600 DD =  ABS (XD):VT = 13:HT =
   26: GOSUB 2650
2610  IF FV = 1 THEN  FLASH
2620 DD =  ABS (YD):VT = 22:HT =
   7: GOSUB 2650
2630 DD =  INT ( SQR (XD * XD + Y
   D * YD)):VT = 22:HT = 21: GOSUB
   2650
2640  VTAB 9: HTAB 34: RETURN
2650  REM COUNTERS
2660  VTAB VT: HTAB HT: PRINT  SPC(
   4);: HTAB HT: PRINT DD;Q$
2670 FT =  INT (DD / 12):IN = DD -
   FT * 12
2680  HTAB HT: PRINT  SPC( 8);: HTAB
   HT: PRINT "("FT"'"IN;Q$")";:
    NORMAL : RETURN
2690  REM H...HARDCOPY
2700  GOSUB 1160: HTAB 13: INVERSE
   : PRINT L$(44): NORMAL : VTAB
   4: HTAB 3: PRINT L$(46): PRINT
   : HTAB 3: PRINT L$(47): PRINT
   : HTAB 3: PRINT L$(67)
2710  VTAB 11: HTAB 10: PRINT L$(
   35): PRINT : HTAB 3: PRINT L
   $(17);
2720  GET C$: IF C$ = " " THEN  GOSUB
   1180: RETURN
2730  IF C$ = "1" THEN N% = 1: GOTO
   2770
2740  IF C$ = "2" THEN N% = 2: GOTO
   2770
2750  IF C$ = "3" THEN 2870
2760  GOTO 2720
2770  VTAB 4: HTAB 3: CALL  - 958
   : PRINT L$(39):H = 8:I2 = 58
   :J = 6:K = 40: GOSUB 2970
2780  VTAB 11: HTAB 3: PRINT L$(4
   8);:DL = 1500: GOSUB 2380: IF
   C$ = "R" THEN  CALL 62450: GOSUB
   1180: GOSUB 4260
2790  GOSUB 1180: PRINT : PRINT D
   $"PR#1": FOR I = 1 TO 3 * N%
   : PRINT : NEXT : PRINT E$"1"
   :TM = 5 + 20 / (N% * N%)
2800  FOR BY = 39 TO 0 STEP  - 1:
    PRINT CR$;: POKE 36,TM:K =
    - 1: FOR TR = 0 TO 80 STEP
   40: FOR OC = 0 TO 896 STEP 1
   28: FOR FL = 0 TO 7168 STEP
   1024:K = K + 1
2810 V%(K) =  PEEK (SA + TR + OC +
   FL + BY): IF N% = 2 THEN  POKE
   768,V%(K): CALL 771:V%(K) =
    PEEK (769):V%(K + 192) =  PEEK
   (770)
2820  NEXT : NEXT : NEXT : FOR H =
   0 TO N% * 192 - 1 STEP 192: IF
   H = 192 THEN  PRINT CR$;: POKE
   36,TM
2830  FOR I = H TO H + 191 STEP 9
   6 / N%: PRINT E$"K" CHR$ (96
   ) CHR$ (0);: FOR J = I TO I -
   1 + 96 / N%: FOR I2 = 1 TO N
   %: POKE 49296,V%(J)
2840  IF  PEEK (49601) < 0 THEN 2
   840
2850  NEXT : NEXT :KC =  PEEK ( -
   16384): IF KC = 160 THEN  POKE
    - 16368,0: PRINT E$"2": PRINT
   D$"PR#0": GOTO 2700
2860  NEXT : NEXT : NEXT : PRINT
   E$"2": PRINT D$"PR#0": RETURN

2870  VTAB 4: HTAB 3: CALL  - 958
   : PRINT L$(32)": ";: GOSUB 1
   120:F$ = IN$: GOSUB 4860: VTAB
   4: HTAB 20: PRINT F$;: CALL
    - 868
2880  ONERR  GOTO 2950
2890 H = 7:I2 = 41:J = 9:K = 40: GOSUB
   2970:VI = 66:VJ = 1
2900  PRINT D$: PRINT D$"BSAVE PI
   C."F$",A$2000,L$2000"
2910  VTAB 4: CALL  - 958
2920 I2 = 60:K = 17: GOSUB 2970:V
   I = 65:VJ = 2: IF C$ <  > "
   " THEN 2920
2930  PRINT D$: PRINT D$"VERIFY A
   RCHITECT.DIR"
2940  GOSUB 1180: RETURN
2950  CALL  - 3288: VTAB 4: CALL
    - 958: HTAB 15: FLASH : PRINT
   " "L$(VI);: NORMAL :DL = 300
   0: GOSUB 2380
2960  HTAB 15: INVERSE : PRINT "
   "L$(VI);: NORMAL : ON VJ GOTO
   2890,2920
2970  REM PROMPTS
2980  VTAB H: HTAB 3: PRINT L$(I2
   ): VTAB J: HTAB 3: PRINT L$(
   K);: GET C$: RETURN
2990  REM I,J,K,M
3000  GOSUB 140:DL = 100: GOSUB 2
   380: GOSUB 140: RETURN
```

```
3010  REM L...LATCH
3020  IF A%(0,1) = 2 THEN  RETURN

3030 ST$ = "L": GOSUB 460: FOR AI
      = 3 TO A%(0,1)
3040  IF XC < A%(AI,2) OR YC < A%
     (AI,3) OR XC > A%(AI,4) OR Y
     C > A%(AI,5) OR A%(AI,0) = 0
      OR A%(AI,1) = 1 THEN 3060
3050 A%(AI,1) = 1: GOSUB 3070
3060  NEXT : GOSUB 490:L = 1: RETURN

3070  REM LATCHED
3080 XS = P5 * (A%(AI,2) + A%(AI,
     4)):YS = P5 * (A%(AI,3) + A%
     (AI,5))
3090  GOSUB 3130: IF A = 1 THEN  XDRAW
     1 AT HS + HR * XS,VS + VR *
     YS: GOTO 3120
3100  IF A = 2 THEN  XDRAW 1 AT H
     S + HR * XS,VS + VR * YS
3110  XDRAW 4 AT HS + HR * XS,VS +
     VR * YS
3120  RETURN
3130  REM LATCH CHECK
3140  IF XS < X1 - 2 THEN XS = X1
      - 2
3150  IF XS > X2 + 2 THEN XS = X2
      + 2
3160  IF YS < Y1 - 2 THEN YS = Y1
      - 2
3170  IF YS > Y2 + 2 THEN YS = Y2
      + 2
3180  RETURN
3190  REM N...NEW
3200  IF N = 0 THEN TI$ = "":TG =
     0: FOR I = 0 TO 5:D(I) = 0:F
     $(I) = " ": NEXT : GOTO 3250

3210  GOSUB 1160: VTAB 12: HTAB 1
     3: PRINT L$(34);
3220  GET C$: IF C$ <  > "Y" AND
     TG = 1 THEN TG = 0: GOSUB 36
     10: RETURN
3230  IF C$ <  > "Y" THEN  GOSUB
     1180: RETURN
3240  HTAB 12: FLASH : PRINT " "L
     $(43);: NORMAL :DL = 1000: GOSUB
     2380:II = A%(0,1): FOR I = 0
      TO II: FOR J = 0 TO 1:A%(I,
     J) = 0: NEXT : NEXT :L = 0
3250  HOME : VTAB 2: HTAB 7: INVERSE
     : PRINT L$(26): NORMAL : VTAB
     5: HTAB 16: PRINT L$(27): VTAB
     7: HTAB 3
3260  PRINT L$(28): VTAB 9: HTAB
     3: PRINT L$(29): VTAB 11: HTAB
     3: PRINT L$(30): VTAB 14: HTAB
     3: PRINT L$(45)":": VTAB 20:
      HTAB 10: PRINT L$(31)
3270  PRINT : HTAB 3: PRINT L$(17
     );: HTAB 29: PRINT "CORD": GOSUB
     3380: GOSUB 3450: GOSUB 3530

3280  IF L1 < W1 THEN L1 = W1:W1 =
     Z
3290 A%(0,0) = 1:A%(0,1) = 2:A%(0
     ,2) = S1
3300 A%(0,3) = 1:A%(0,4) = L1:A%(
     0,5) = W1
3310 A%(1,0) = 2:A%(2,0) = 2:A%(1
     ,1) = 1:A%(2,1) = 0
3320 A%(1,2) = WS:A%(1,3) = WS
3330 A%(1,4) = L1 + WS:A%(1,5) =
     W1 + WS
3340 A%(2,2) = 0:A%(2,3) = 0
3350 A%(2,4) = L1 + 2 * WS:A%(2,5
     ) = W1 + 2 * WS
3360 A%(0,6) = 0:A%(0,7) = 0:A%(1
     ,6) = 1:A%(1,7) = 0:A%(2,6) =
     1:A%(2,7) = 0
3370  GOSUB 3610: RETURN
3380  REM CHECK
3390 W1 = D(1) + 12 * D(0):L1 = D
     (3) + 12 * D(2):S1 = D(5) +
     12 * D(4):Z = L1
3400 FM = 1: IF W1 = 0 OR L1 = 0 OR
     S1 = 0 THEN FM = 0
3410  RETURN
3420  REM CONVERT
3430 FT =  INT (D(K + 1) / 12):D(
     K) = D(K) + FT:D(K + 1) = D(
     K + 1) - 12 * FT
3440 F$(K) =  STR$ (D(K)):F$(K +
     1) =  STR$ (D(K + 1)): RETURN

3450  REM DIMENSIONS
3460  FOR J = 0 TO 2:K = 2 * J
3470  IF F$(K) = "--" OR F$(K + 1
     ) = "--" THEN 3500
3480  GOSUB 3420
3490  IF D(K) > 99 THEN F$(K) = "
     **":F$(K + 1) = "**":FM = 0:
     D(K) = 0:D(K + 1) = 0
3500  VTAB 7 + K: HTAB 18: PRINT
      RIGHT$ (" " + F$(K),2);
3510  HTAB 28: PRINT  RIGHT$ (" "
      + F$(K + 1),2);: IF G = 1 THEN
      RETURN
3520  NEXT : VTAB 14: HTAB 13: PRINT
     TI$: VTAB 22: HTAB 36: RETURN

3530  REM MODIFY
3540  GET C$: IF C$ = " " AND FM =
     1 THEN  RETURN
3550  IF C$ = " " THEN  CALL 6447
     7
3560 CS =  ASC (C$) - 65: IF CS <
     0 OR CS > 6 THEN 3540
3570  IF CS = 6 THEN  GOSUB 6110:
      GOTO 3540
3580 F$(CS) = "--": GOSUB 3450: VTAB
     17: HTAB 3: PRINT L$(33)": "
     ;
3590  GOSUB 1120: VTAB 17: CALL  -
     868
3600 D(CS) =  ABS ( VAL (IN$)):F$
     (CS) = " ": GOSUB 3380: GOSUB
     3450: GOTO 3540
3610  REM OUTLINE
3620 MV = 0:FH = 0:FV = 0:FL = 0:
     ST = 2:N = 1: GOSUB 1330
3630 XC = P5 * (X1 + X2):YC = P5 *
     (Y1 + Y2):XF = XC:YF = YC
3640  HCOLOR= 3: POKE 230,32: CALL
     62450: POKE  - 16297,0: POKE
      - 16302,0: POKE  - 16304,0:
      GOSUB 4260: RETURN
3650  REM CLEAR
3660  VTAB 22: HTAB 1: CALL  - 86
     8: VTAB 21: CALL  - 868: RETURN

3670  REM FILES
3680  POKE 34,5: POKE 35,20: VTAB
     6: HTAB 1: FOR I = 1 TO DF STEP
     3: FOR J = I TO I + 2: IF J >
     DF THEN 3730
3690  PRINT " ";: IF J = TI THEN
      INVERSE
3700  IF J = TF THEN  FLASH
3710  PRINT  RIGHT$ (" " +  STR$
     (J) + " ",3) LEFT$ (DF$(J) +
     "         ",9);: NORMAL : REM
     9 SPACES
3720  NEXT : PRINT " ";: NEXT
3730  POKE 34,0: POKE 35,24: VTAB
     24: HTAB 36 + (N = 0) * 2: RETURN

3740  REM PROMPT
3750  IF DF = 0 THEN K = 63: GOSUB
     3950
3760  GOSUB 3650: HTAB 3: PRINT L
     $(33)": "S$;: GOSUB 1120:TF =
      VAL (IN$)
3770  IF TF < 1 OR TF > DF THEN K
      = 59: GOSUB 3950
3780  GOSUB 3670: IF C$ <  > "D" AND
     C$ <  > "R" THEN  RETURN
3790  VTAB 22: HTAB 13: INVERSE :
      PRINT L$(34);: NORMAL : GET
     B$
3800  IF B$ <  > "Y" THEN  POP : GOTO
     4640
3810  IF C$ = "R" OR C$ = "Q" THEN
      RETURN
3820 M$ = "A." + DF$(TF): PRINT D
     $: PRINT D$"DELETE "M$: RETURN

3830  REM PROMPT2
3840  GOSUB 3650: IF N = 0 AND TG
      = 0 THEN K = 62: GOSUB 3950
3850  IF C$ = "S" THEN 3900
3860  IF TI = 0 THEN 3740
3870 TF = TI: GOSUB 3650: GOSUB 3
     670: VTAB 21: HTAB 3: PRINT
     L$(33)": "S$TI: HTAB 13: INVERSE
     : PRINT L$(34);: NORMAL : GET
     B$
3880  IF B$ <  > "Y" THEN  POP : GOTO
     4640
3890  RETURN
3900  GOSUB 3650: VTAB 21: HTAB 3
     : PRINT L$(57)": "S$;: GOSUB
     1120:F$ =  LEFT$ (IN$,9): GOSUB
     4860: VTAB 21: HTAB 20: PRINT
     F$;: CALL  - 868: PRINT
3910  IF TM <  > 0 THEN  HTAB 3: INVERSE
     : PRINT L$(64);: NORMAL : GET B$
     : IF B$ = "Y" THEN TF = TM: GOSUB
     3670: RETURN
3920  IF TM <  > 0 THEN  POP : GOTO
     4640
3930  IF DF = 42 THEN K = 61: GOSUB
     3950
3940 DF = DF + 1:DF$(DF) = F$:TF =
     DF: GOSUB 3670: RETURN
3950  REM MESSAGES
3960  IF K = 66 THEN  PRINT D$: PRINT
     D$"CLOSE"
3970  VTAB 22: HTAB 1: CALL  - 86
     8:HT =  INT ((39 -  LEN (L$(
     K))) / 2): HTAB HT: INVERSE
     : PRINT L$(K);: NORMAL :TF =
     0: GOSUB 3670
3980  POP : POP : GOTO 4530
3990  REM DELETE
4000  IF TF = DF THEN 4020
4010  FOR I = TF TO DF - 1:DF$(I)
      = DF$(I + 1): NEXT
4020  IF TI = TF THEN TI = 0
4030  IF TI > TF THEN TI = TI - 1

4040 DF = DF - 1: RETURN
4050  REM P...PLACE
4060  IF L = 0 OR MV = 0 OR A%(0,
     1) = 2 THEN  RETURN
4070 ST$ = "P": GOSUB 460: GOSUB
     1200:XH = XC - XF:YH = YC -
     YF
4080  FOR AI = 3 TO A%(0,1):OB =
     A%(AI,0)
4090  IF A%(AI,1) = 0 THEN 4190
4100  IF (OB = 3 OR OB = 4) AND  FN
     OT(XF) = 0 THEN 4190
4110  IF (OB = 2 OR OB = 5) AND  FN
     IN(XF) = 0 THEN 4190
4120 XE = XH:YE = YH
4130  IF (OB = 3 OR OB = 4) AND  FN
     HO(AI) THEN YE = 0: IF YF >
     WS AND YF < WS + W1 THEN 419
     0
4140  IF (OB = 3 OR OB = 4) AND  FN
     VE(AI) THEN XE = 0: IF XF >
     WS AND XF < WS + L1 THEN 419
     0
4150  HCOLOR= 0:CR = 3:CL = 0: GOSUB
     4330
4160 A%(AI,2) = A%(AI,2) + XE:A%(
     AI,3) = A%(AI,3) + YE
4170 A%(AI,4) = A%(AI,4) + XE:A%(
     AI,5) = A%(AI,5) + YE
4180  HCOLOR= 3:CR = 0:CL = 3: GOSUB
     4330
4190  NEXT :XF = XC:YF = YC: GOSUB
     1860: RETURN
4200  REM Q...QUIT
4210  GOSUB 1160: VTAB 12: HTAB 1
     3: PRINT L$(34);
4220  GET C$: IF C$ = "Y" THEN  POKE
     106,8: POKE 105,0: HOME : END

4230  GOSUB 1180: RETURN
4240  REM R...REDRAW
4250  CALL 62450: GOSUB 1160: VTAB
     12: HTAB 10: FLASH : PRINT "
      "L$(50);: NORMAL :DL = 1000
     : GOSUB 2380: GOSUB 4370: GOSUB
     1180
4260  IF A%(0,3) = 1 THEN  GOSUB
     1410
```

```
4270 FOR AI = 1 TO A%(0,1)
4280 GOSUB 4330: NEXT : IF A%(1,
     1) = 0 THEN 4310
4290 IV = 0:VL = 1:VT = 1:HT = 40
     :ST$ = L$(49): GOSUB 640
4300 VL = 0:VT = 24: IF TI$ < >
     "" THEN HT = 1 + INT ((35 -
     LEN (TI$)) / 2):ST$ = TI$: GOSUB
     640
4310 GOSUB 490
4320 GOSUB 1200: RETURN
4330 REM DRAW OBJECTS
4340 IF A%(AI,0) > = 2 AND A%(A
     I,0) < = 5 THEN GOSUB 1220
4350 IF AI > 2 AND A%(AI,1) = 1 THEN
     GOSUB 3070
4360 RETURN
4370 REM HOUSKEEPING
4380 X = FRE (0)
4390 AN = A%(0,1):AI = 2: IF AN =
     2 THEN RETURN
4400 FOR I = 3 TO AN
4410 IF A%(I,0) = 0 THEN 4460
4420 IF A%(I,1) = 0 AND ( FN VH(
     2) * FN VH(4) OR FN RH(2) *
     FN RH(4) OR FN VH(3) * FN
     VH(5) OR FN BV(3) * FN BV(
     5)) THEN 4460
4430 AI = AI + 1: IF AI = I THEN
     4460
4440 FOR J = 0 TO 7
4450 A%(AI,J) = A%(I,J): NEXT
4460 NEXT :A%(0,1) = AI: IF AI =
     2 THEN AI = 3
4470 RETURN
4480 REM S...SAVE
4490 ONERR GOTO 4800
4500 GOSUB 1160: INVERSE : HTAB
     12: PRINT L$(54): NORMAL : PRINT
     : HTAB 3: PRINT L$(55): HTAB
     3: PRINT L$(56): VTAB 24: HTAB
     3: PRINT L$(17);
4510 IF N = 0 THEN HTAB 3: PRINT
     "PRESS COMMAND KEY OR SPACE
     BAR...";
4520 GOSUB 3670
4530 GET C$
4540 IF C$ = " " AND TG = 1 THEN
     TG = 0: GOSUB 3610: RETURN
4550 IF C$ = " " AND N = 0 THEN
     CA = 78: POP : GOTO 370
4560 IF C$ = " " THEN GOSUB 118
     0: RETURN
4570 IF C$ = "N" THEN CA = 78: POP
     : GOTO 370
4580 IF C$ = "Q" THEN GOSUB 379
     0: HOME : END
4590 IF C$ = "D" THEN S$ = "DELE
     TE ": GOSUB 3740: GOSUB 3990
     : GOSUB 4810: POKE 34,5: POKE
     35,20: CALL - 936: GOTO 464
     0
4600 IF C$ = "G" THEN S$ = "GET
     ": GOSUB 3740: GOSUB 4650:TG
     = 1: GOTO 4640
4610 IF C$ = "S" THEN S$ = "SAVE
     ": GOSUB 3830: GOSUB 4750: GOSUB
     4810: GOTO 4640
4620 IF C$ = "R" THEN S$ = "REVI
     SE ": GOSUB 3830: GOSUB 4750
     : GOTO 4640
4630 GOTO 4530
4640 X = FRE (0):TF = 0: GOSUB 3
     650: GOSUB 3670: GOTO 4530
4650 REM GET
4660 M$ = "A." + DF$(TF): VTAB 22
     : PRINT D$: PRINT D$"OPEN "M
     $: PRINT D$"READ "M$
4670 GOSUB 1120:TI$ = MID$ (IN$
     ,1)
4680 FOR I = 0 TO 7: INPUT A%(0,
     I): NEXT
4690 FOR I = 1 TO A%(0,1): FOR J
     = 0 TO 7: INPUT A%(I,J): NEXT
     : NEXT
4700 W1 = A%(0,5):L1 = A%(0,4):S1
     = A%(0,2):TI = TF:D(0) = 0:
     D(2) = 0:D(4) = 0:D(1) = W1:
     D(3) = L1:D(5) = S1
4710 GOSUB 1330: FOR J = 0 TO 2:
     K = 2 * J: GOSUB 3420: NEXT

4720 PRINT D$"CLOSE "M$:L = 0: IF
     A%(0,1) = 2 THEN RETURN
4730 FOR I = 3 TO A%(0,1): IF A%
     (I,1) = 1 THEN L = 1: RETURN

4740 NEXT : RETURN
4750 REM PUT
4760 M$ = "A." + DF$(TF): VTAB 22
     : PRINT D$: PRINT D$"OPEN "M
     $: PRINT D$"CLOSE": PRINT D$
     "DELETE "M$
4770 PRINT D$"OPEN "M$: PRINT D$
     "WRITE "M$
4780 PRINT TI$: FOR I = 0 TO A%(
     0,1): FOR J = 0 TO 7: PRINT
     A%(I,J): NEXT : NEXT :TI =
     F
4790 PRINT D$"CLOSE "M$: RETURN

4800 CALL - 3288:K = 66: GOSUB
     3950
4810 REM DIRECTORY
4820 M$ = "ARCHITECT.DIR": PRINT
     D$"OPEN "M$: PRINT D$"CLOSE"
     : PRINT D$"DELETE "M$
4830 PRINT D$"OPEN "M$: PRINT D$
     "WRITE "M$
4840 DF$(0) = STR$ (DF): FOR I =
     0 TO DF: PRINT DF$(I): NEXT

4850 PRINT D$"CLOSE "M$: RETURN

4860 REM MATCH
4870 IF RIGHT$ (F$,1) = " " THEN
     F$ = MID$ (F$,1, LEN (F$) -
     1): GOTO 4870
4880 Z$ = "": FOR I = 1 TO LEN (
     F$):T$ = MID$ (F$,I,1): IF
     T$ < > "," AND T$ < > ":" THEN
     Z$ = Z$ + T$
4890 NEXT :F$ = Z$
4900 TM = 0: FOR I = 1 TO DF: IF
     F$ = DF$(I) THEN TM = I:I =
     DF
4910 NEXT : RETURN
4920 REM T...TURN
4930 IF L = 0 OR FN IN(XF) = 0 OR
     A%(0,1) = 2 THEN RETURN
4940 ST$ = "T": GOSUB 460: GOSUB
     1200:XD = XC - XF:YD = YC -
     YF:XE = XF:YE = YF
4950 FOR AI = 3 TO A%(0,1):OB =
     A%(AI,0)
4960 IF A%(AI,1) = 0 OR OB = 3 OR
     OB = 4 OR (OB = 5 AND (XD <
     > 0 OR YD < > 0)) THEN 507
     0
4970 HCOLOR= 0: GOSUB 4330: IF X
     D = 0 AND YD = 0 THEN 5000
4980 A%(AI,6) = XD:A%(AI,7) = YD:
     IF XD = 0 THEN XE = P5 * (A
     %(AI,2) + A%(AI,4)):YE = P5 *
     (A%(AI,3) + A%(AI,5)):A%(AI,
     6) = 1:A%(AI,7) = 0: GOTO 50
     00
4990 GOTO 5060
5000 XL = XE - YE + A%(AI,3):YT =
     YE + XE - A%(AI,2)
5010 XR = XE - YE + A%(AI,5):YB =
     YE + XE - A%(AI,4)
5020 A%(AI,2) = XL:A%(AI,4) = XR
5030 IF XL > XR THEN A%(AI,2) =
     XR:A%(AI,4) = XL
5040 A%(AI,3) = YT:A%(AI,5) = YB
5050 IF YT > YB THEN A%(AI,3) =
     YB:A%(AI,5) = YT
5060 HCOLOR= 3: GOSUB 4330
5070 NEXT : GOSUB 490: GOSUB 120
     0: RETURN
5080 REM ROTATE RECTANGLE
5090 XN = P5 * (XL + XR):YX = P5 *
     (YT + YB):CX = A%(AI,6):SI =
     A%(AI,7)
5100 DD = SQR (CX * CX + SI * SI
     ):CX = CX / DD:SI = SI / DD
5110 XX = P5 * (XR - XL):YY = P5 *
     (YT - YB)
5120 XA = XN + XX * CX - YY * SI:
     YA = YX + XX * SI + YY * CX
5130 XB = XN + XX * CX + YY * SI:
     YB = YX + XX * SI - YY * CX
5140 XU = XN - XX * CX + YY * SI:
     YU = YX - XX * SI - YY * CX
5150 XV = XN - XX * CX - YY * SI:
     YV = YX - XX * SI + YY * CX
5160 XA = FN BB(XA):XB = FN BB(
     XB):XU = FN BB(XU):XV = FN
     BB(XV)
5170 YA = FN BD(YA):YB = FN BD(
     YB):YU = FN BD(YU):YV = FN
     BD(YV)
5180 XA = HS + HR * XA:XB = HS +
     HR * XB:XU = HS + HR * XU:XV
     = HS + HR * XV:YA = VS + VR
     * YA:YB = VS + VR * YB:YU =
     VS + VR * YU:YV = VS + VR *
     YV
5190 HPLOT XA,YA TO XB,YB TO XU,
     YU TO XV,YV TO XA,YA: RETURN

5200 REM U...UNLATCH
5210 IF A%(0,1) = 2 THEN RETURN

5220 ST$ = "U": GOSUB 460: FOR AI
     = 3 TO A%(0,1)
5230 IF A%(AI,1) = 1 THEN GOSUB
     5250:A%(AI,1) = 0
5240 NEXT : GOSUB 490:L = 0: RETURN

5250 REM UNLATCHED
5260 XS = P5 * (A%(AI,2) + A%(AI,
     4)):YS = P5 * (A%(AI,3) + A%
     (AI,5))
5270 GOSUB 3130: XDRAW 4 AT HS +
     HR * XS,VS + VR * YS: RETURN

5280 REM V...VERIFY
5290 GOSUB 1160: INVERSE : HTAB
     10: PRINT L$(25)
5300 VTAB 3: HTAB 4: PRINT L$(36
     );: HTAB 14: PRINT L$(37);: HTAB
     28: PRINT L$(38): NORMAL
5310 VT = 5:VI = 20: VTAB VT: HTAB
     4: PRINT L$(VI)
5320 HT = 14:DD = L1: GOSUB 2650
5330 HT = 28:DD = W1: GOSUB 2650
5340 IF L = 0 THEN 5420
5350 VJ = 1: FOR I = 3 TO A%(0,1)

5360 IF A%(I,1) = 0 THEN 5410
5370 VT = VT + 3:VI = A%(I,0) + 1
     9: VTAB VT: HTAB 4: PRINT L$
     (VI)
5380 DD = A%(I,4) - A%(I,2):HT =
     14: GOSUB 2650
5390 DD = A%(I,5) - A%(I,3):HT =
     28: GOSUB 2650
5400 VJ = VJ + 1: IF VJ > 5 THEN
     5420
5410 NEXT
5420 VTAB 24: HTAB 3: PRINT L$(1
     7);
5430 GET C$: IF C$ = " " THEN GOSUB
     1180: RETURN
5440 GOTO 5430
5450 REM W...WINDOW
5460 IF FN OT(XF) = 0 OR MV = 0
     THEN RETURN
5470 IF A = 0 THEN ST$ = "W": GOSUB
     1820:A%(AI,0) = 3
5480 A6 = 0:A7 = 0: GOSUB 2120: GOSUB
     1220: GOSUB 1860: RETURN
5490 REM BAR
5500 IF A%(AI,6) < > 0 THEN HPLOT
     LX,TY TO LX,BY: HPLOT RX,TY TO
     RX,BY:TY = P5 * (TY + BY):BY
     = TY
5510 IF A%(AI,7) < > 0 THEN HPLOT
     LX,TY TO RX,TY: HPLOT LX,BY TO
     RX,BY:LX = P5 * (LX + RX):RX
     = LX
5520 HPLOT LX,TY TO RX,BY: RETURN

5530 REM X...EXIT W/O DOOR
5540 F = 1: GOSUB 2050:F = 0: RETURN

5550 REM DOORWAY
5560 HCOLOR= CR:MX = 262:MY = 17
     2
5570 IF FN HO(AI) THEN LX = LX +
     1:LX = FN BB(LX):RX = RX -
     1:RX = FN BB(RX): HPLOT LX,
     TY TO RX,TY: HPLOT LX,BY TO
     RX,BY
5580 IF FN VE(AI) THEN TY = TY +
     1:TY = FN BD(TY):BY = BY -
     1:BY = FN BD(BY): HPLOT LX,
     TY TO LX,BY: HPLOT RX,TY TO
     RX,BY
5590 HCOLOR= CL
5600 IF A%(AI,6) = 0 THEN 5670
5610 D = XR - XL:CI = A%(AI,6):TI
```

```
    = 180 * PR:T2 = 360 * PR
5620  IF CI = 1 THEN XG = XL:YG =
    YB: GOSUB 2220
5630  IF CI = 2 THEN XG = XR:YG =
    YB: GOSUB 2220
5640 T2 = T1:T1 = 0
5650  IF CI = 3 THEN XG = XR:YG =
    YT: GOSUB 2220
5660  IF CI = 4 THEN XG = XL:YG =
    YT: GOSUB 2220
5670  IF A%(AI,7) = 0 THEN  RETURN

5680 D = YB - YT:CI = A%(AI,7):T1
     =  - 90 * PR:T2 = 90 * PR
5690  IF CI = 1 THEN XG = XR:YG =
    YT: GOSUB 2220
5700  IF CI = 4 THEN XG = XR:YG =
    YB: GOSUB 2220
5710 T1 = T2:T2 = 270 * PR
5720  IF CI = 2 THEN XG = XL:YG =
    YT: GOSUB 2220
5730  IF CI = 3 THEN XG = XL:YG =
    YB: GOSUB 2220
5740  RETURN
5750  REM *...CIRCLE
5760  IF MV = 0 THEN  RETURN
5770  IF A = 0 THEN ST$ = "*": GOSUB
    1820:A%(AI,0) = 5
5780 XD = XC - XF:YD = YC - YF:D =
     INT ( SQR (XD * XD + YD * Y
    D))
5790 A%(AI,2) = XF - D:A%(AI,4) =
    XF + D
5800 A%(AI,3) = YF - D:A%(AI,5) =
    YF + D
5810 A%(AI,6) = 0:A%(AI,7) = 0
5820  GOSUB 1220: GOSUB 1860: RETURN

5830  REM #...GRID & TITLE
5840 G$(0) = "ON":G$(1) = "OFF":G
    $(2) = "ON":G$(3) = "OFF":FM
     = 0:G = 1:E(0) = A%(0,3):E(
    1) = A%(1,1)
5850  GOSUB 1160: HTAB 11: INVERSE
    : PRINT L$(51): NORMAL : VTAB
    4: HTAB 3: PRINT L$(52): PRINT
    : HTAB 3: PRINT L$(53)
5860  VTAB 9: HTAB 16: PRINT L$(2
    7): PRINT : HTAB 3: PRINT L$
    (30): VTAB 14: HTAB 3: PRINT
    L$(45)":": VTAB 20: HTAB 10:
     PRINT L$(31): PRINT : HTAB
    3: PRINT L$(17)
5870  GOSUB 6020
5880  GET C$: IF C$ = " " AND FM =
    0 THEN  GOSUB 1180:G = 0: RETURN

5890  IF C$ = " " AND S1 = 0 THEN
     CALL 64477: GOTO 5880
5900  IF C$ = " " THEN  CALL 6245
    0: GOSUB 1180:G = 0: GOSUB 4
    260: RETURN
5910 CC =  ASC (C$) - 65: IF CC <
    0 OR CC > 6 THEN 5880
5920 FM = 1: IF CC = 0 AND A%(0,3
    ) = 0 THEN A%(0,3) = 1:E(0) =
    1: GOTO 6010
5930  IF CC = 1 AND A%(0,3) = 1 THEN
    A%(0,3) = 0:E(0) = 0: GOTO 6
    010
5940  IF CC = 2 AND A%(1,1) = 0 THEN
    A%(1,1) = 1:E(1) = 1: GOTO 6
    010
5950  IF CC = 3 AND A%(1,1) = 1 THEN
    A%(1,1) = 0:E(1) = 0: GOTO 6
    010
5960  IF CC >  = 0 AND CC <  = 3 THEN
    5880
5970  IF CC = 6 THEN  GOSUB 6110:
     GOTO 5880
5980 F$(CC) = "--":K = 4: GOSUB 3
    470: VTAB 17: HTAB 3: PRINT
    L$(33)": ":
5990  GOSUB 1120: VTAB 17: CALL  -
    868:D(CC) =  ABS ( VAL (IN$)
    ):F$(CC) = " ": GOSUB 3470:F
    M = 1
6000 S1 = D(5) + 12 * D(4):HD = S
    1 * HR:VD = S1 * VR: VTAB 22
    : HTAB 36: GOTO 5880
6010  GOSUB 6020: GOTO 5880
6020  REM ENTRIES
6030  FOR J = 0 TO 1:K = 2 * J: VTAB
     4 + K: HTAB 18
6040  IF E(J) = 1 THEN  INVERSE
6050  PRINT G$(K);: HTAB 28: NORMAL

6060  IF E(J) = 0 THEN  INVERSE
6070  PRINT G$(K + 1);: NORMAL : NEXT

6080 K = 4: GOSUB 3470
6090  VTAB 14: HTAB 13: PRINT TI$

6100  VTAB 22: HTAB 36: RETURN
6110  REM TITLE
6120  VTAB 14: HTAB 13: PRINT  SPC(
    35): VTAB 14: HTAB 13: GOSUB
    1120:TI$ =  LEFT$ (IN$,35): VTAB
    14: HTAB 13: PRINT TI$;: CALL
     - 868: VTAB 22: HTAB 36: RETURN
6130  REM <...SMALLER STEP
6140  IF ST > 1 THEN ST = ST - 1
6150  RETURN
6160  REM >...LARGER STEP
6170  IF ST < 3 THEN ST = ST + 1
6180  RETURN
6190  REM ?...HELP
6200  GOSUB 1160: INVERSE : PRINT
    L$(0): NORMAL : PRINT : GOSUB
    740: PRINT : PRINT : PRINT L
    $(1): PRINT L$(2): PRINT : PRINT
    : PRINT L$(17);
6210  GET C$: IF C$ = " " THEN  GOSUB
    1180: RETURN
6220  GOTO 6210
```

LISTING #2

```
10  REM  ************************

20  REM  *     MENU.CREATE       *

30  REM  *  BY    COMPUHOUSE     *

40  REM  *  COPYRIGHT (C) 1985   *

50  REM  *  BY COMPUTING  AGE    *

70  REM  ************************

80  HOME : VTAB 12: PRINT "INITIA
    LIZING ..."
90 NL = 67: DIM L$(NL)
100 Q$ =  CHR$ (34)
110  REM CURSOR DEFINITION
120  DATA 4,0,10,0,18,0,29,0,39,0

130  DATA 32,141,58,63,183,9,36,0

140  DATA 32,108,145,58,63,63,183
    ,74,33,36,0
150  DATA 12,12,223,115,22,23,77,
    225,7,0
160  DATA 9,32,255,55,22,46,13,37
    ,4,0
170  DATA 232,64,233,3
180  REM HI-RES EXPAND
190  DATA 0,0,0,162,0,142,1,3,142
    ,2,3,189,43,3,45,0,3,240,18,
    189,50,3,13,1,3,141,1,3,189,
    57,3,13,2,3,141,2,3,232,224,
    7,48,225,96,64,32,16,8,4,2,1
    ,96,24,6,1,0,0,0,0,0,0,64,48
    ,12,3
200 L$(0) = "FUNCTIONS ASSIGNED T
    O THE KEYBOARD:"
210 L$(1) = "'I', 'J', 'K', 'M',
    '<', AND '>' CONTROL"
220 L$(2) = "THE MOVEMENT OF THE
    CURSOR."
230 L$(3) = "*********************
    *********************"
240 L$(4) = "*             PROFESSIONA
    L ARCHITECT          *"
250 L$(5) = "* COPYRIGHT (C) 1985
     BY COMPUTING  AGE *"
260 L$(6) = "A-ALTER         K-RIGHT
         U-UNLATCH          "
270 L$(7) = "B-BOX           L-LATCH
         V-VERIFY           "
280 L$(8) = "C-COPY          M-DOWN
         W-WINDOW           "
290 L$(9) = "D-DOOR          N-NEW
         X-EXIT W/O DOOR "
300 L$(10) = "E-ERASE        O-OLD
         Y-YES              "
310 L$(11) = "F-FIX          P-PLAC
    E    *-CIRCLE           "
320 L$(12) = "G-GET DATA     Q-QUIT
         #-GRID & TITLE     "
330 L$(13) = "H-HARDCOPY     R-REDR
    AW   <-SMALLER STEP     "
340 L$(14) = "I-UP           S-SAVE
         >-LARGER STEP      "
350 L$(15) = "J-LEFT         T-TURN
         ?-HELP             "
360 L$(16) = "PRESS 'N' OR 'O' TO
     CONTINUE..."
370 L$(17) = "PRESS THE 'SPACEBAR
    ' TO RETURN..."
380 L$(18) = "TYPE  ID   XL   YT
      XR   YB   COS  SIN"
390 L$(19) = "*** DISTANCE BETWEE
    N CURSORS ***"
400 L$(20) = "ROOM"
410 L$(21) = "BOX"
420 L$(22) = "WINDOW"
430 L$(23) = "DOOR"
440 L$(24) = "CIRCLE"
450 L$(25) = "*** MEASUREMENTS **
    *"
460 L$(26) = "*** STARTING A NEW
     ROOM ***"
470 L$(27) = "FEET      INCHES"
480 L$(28) = "WIDTH         A)
        B)"
490 L$(29) = "LENGTH        C)
        D)"
500 L$(30) = "GRID SIZE   E)
        F)"
510 L$(31) = "SELECT (A-G) TO MOD
    IFY"
520 L$(32) = "ENTER FILE NAME"
530 L$(33) = "ENTER NUMBER"
540 L$(34) = "ARE YOU SURE?"
550 L$(35) = "SELECT (1-3)"
560 L$(36) = "OBJECT"
570 L$(37) = "HORIZONTAL"
580 L$(38) = "VERTICAL"
590 L$(39) = "TURN PRINTER ON AND
     POSITION PAPER"
600 L$(40) = "PRESS ANY KEY WHEN
    READY..."
610 L$(41) = "INSERT NEW DISK IN
    DRIVE"
620 L$(42) = "OUT OF BUFFER SPACE
    "
630 L$(43) = "INITIALIZING "
640 L$(44) = "*** HARDCOPY ***"
650 L$(45) = "G) TITLE"
660 L$(46) = "1) 3" + Q$ + " X 4"
     + Q$
670 L$(47) = "2) 6" + Q$ + " X 8"
     + Q$
680 L$(48) = "PRESS 'SPACEBAR' TO
     STOP PRINTING..."
690 L$(49) = "PROFESSIONAL ARCHIT
    ECT "
700 L$(50) = "REORGANIZING BUFFER
    "
710 L$(51) = "*** GRID & TITLE **
    *"
720 L$(52) = "GRID          A)
        B)"
730 L$(53) = "TITLE         C)
        D)"
740 L$(54) = "*** DISK FILES ***"

750 L$(55) = "D-DELETE       N-NEW
         R-REVISE"
760 L$(56) = "G-GET          Q-QUIT
         S-SAVE"
770 L$(57) = "ENTER NAME"
780 L$(58) = "('R' WILL REDRAW BE
    FORE PRINTING)"
790 L$(59) = "OUT OF RANGE"
800 L$(60) = "REPLACE 'ARCHITECT'
     DISK IN DRIVE"
810 L$(61) = "MAXIMUM EXCEEDED"
820 L$(62) = "NO DATA"
830 L$(63) = "NO FILES"
840 L$(64) = "FILE EXISTS - DO YO
    U WISH TO REVISE?"
850 L$(65) = "WRONG DISK "
860 L$(66) = "DISK ERROR "
870 L$(67) = "3) OUTPUT TO DISK"
```

專業室內設計師

```
880  REM CREATE ARCHITECT.MENU
890 D$ =  CHR$ (4): REM  CTRL-D
900  PRINT D$"OPEN ARCHITECT.MENU
     "
910  PRINT D$"CLOSE"
920  PRINT D$"DELETE ARCHITECT.ME
     NU"
930  PRINT D$"OPEN ARCHITECT.MENU
     "
940  PRINT D$"WRITE ARCHITECT.MEN
     U"
950  FOR I = 0 TO NL
960  PRINT L$(I): NEXT
970  FOR I = 1 TO 117
980  READ X: PRINT X: NEXT
990  PRINT D$"CLOSE ARCHITECT.MEN
     U"
1000  HOME : VTAB 12: PRINT "DONE
     ": END ■
```

LISTING #3

```
10  REM  *********************
20  REM  *    ALPH.CREATE    *
30  REM  * COPYRIGHT(C) 1985 *
40  REM  * BY COMPUTING  AGE *
50  REM  *********************
100  HOME : VTAB 12: PRINT "CREAT
     ING HI-RES CHARACTER SHAPE T
     ABLE..."
110 ADDR = 0:LN = 0
120  DIM AL%(64,8)
130  FOR I = 0 TO 64
140  FOR J = 0 TO 7
150  READ AL%(I,J)
160  NEXT J,I
170 ADDR =  PEEK (107) + 256 *  PEEK
     (108)
180 LN =  PEEK (109) + 256 *  PEEK
     (110) - ADDR
190  PRINT  CHR$ (4);"BSAVE ARCHI
     TECT.ALPH, A";ADDR;",L";LN
200  HOME : VTAB 12: PRINT "DONE"
     : END
210  DATA 0,0,0,0,0,0,0,0: REM  S
     PACE
220  DATA 0,8,8,8,8,8,0,8: REM  !

230  DATA 0,20,20,20,0,0,0,0: REM
      "
240  DATA 0,20,20,62,20,62,20,20:
      REM  #
250  DATA 0,8,60,10,28,40,30,8: REM
      $
260  DATA 0,6,38,16,8,4,50,48: REM
      %
270  DATA 0,4,10,10,4,42,18,44: REM
      &
280  DATA 0,8,8,8,0,0,0,0: REM
      '
290  DATA 0,8,4,2,2,2,4,8: REM  (

300  DATA 0,8,16,32,32,32,16,8: REM
      )
310  DATA 0,8,42,28,8,28,42,8: REM
      *
320  DATA 0,0,8,8,62,8,8,0: REM
      +
330  DATA 0,0,0,0,0,8,8,4: REM  ,

340  DATA 0,0,0,0,62,0,0,0: REM
     -
350  DATA 0,0,0,0,0,0,0,8: REM  .

360  DATA 0,0,32,16,8,4,2,0: REM
      /
370  DATA 0,28,34,50,42,38,34,28:
      REM  0
380  DATA 0,8,12,8,8,8,8,28: REM
      1
390  DATA 0,28,34,32,24,4,2,62: REM
      2
400  DATA 0,62,32,16,24,32,34,28:
      REM   3
410  DATA 0,16,24,20,18,62,16,16:
      REM  4
420  DATA 0,62,2,30,32,32,34,28: REM
      5
430  DATA 0,56,4,2,30,34,34,28: REM
      6
440  DATA 0,62,32,16,8,4,4,4: REM
      7
450  DATA 0,28,34,34,28,34,34,28:
      REM  8
460  DATA 0,28,34,34,60,32,16,14:
      REM  9
470  DATA 0,0,0,8,0,8,0,0: REM  :

480  DATA 0,0,0,8,0,8,8,4: REM  ;

490  DATA 0,16,8,4,2,4,8,16: REM
      <
500  DATA 0,0,0,62,0,62,0,0: REM
      =
510  DATA 0,4,8,16,32,16,8,4: REM
      >
520  DATA 0,28,34,16,8,8,0,8: REM
      ?
530  DATA 0,28,34,42,58,26,2,60: REM
        @
540  DATA 0,8,20,34,34,62,34,34: REM
      A
550  DATA 0,30,34,34,30,34,34,30:
      REM  B
560  DATA 0,28,34,2,2,2,34,28: REM
      C
570  DATA 0,30,34,34,34,34,34,30:
      REM  D
580  DATA 0,62,2,2,30,2,2,62: REM
      E
590  DATA 0,62,2,2,30,2,2,2: REM
      F
600  DATA 0,60,2,2,2,50,34,60: REM
      G
610  DATA 0,34,34,34,62,34,34,34:
      REM  H
620  DATA 0,28,8,8,8,8,8,28: REM
      I
630  DATA 0,32,32,32,32,32,34,28:
      REM  J
640  DATA 0,34,18,10,6,10,18,34: REM
      K
650  DATA 0,2,2,2,2,2,2,62: REM
     L
660  DATA 0,34,54,42,34,34,34,34:
      REM  M
670  DATA 0,34,34,38,42,50,34,34:
      REM  N
680  DATA 0,28,34,34,34,34,34,28:
      REM  O
690  DATA 0,30,34,34,30,2,2,2: REM
      P
700  DATA 0,28,34,34,34,42,18,44:
      REM   Q
710  DATA 0,30,34,34,30,10,18,34:
      REM  R
720  DATA 0,28,34,2,28,32,34,28: REM
      S
730  DATA 0,62,8,8,8,8,8,8: REM
     T
740  DATA 0,34,34,34,34,34,34,28:
      REM  U
750  DATA 0,34,34,34,34,34,20,8: REM
      V
760  DATA 0,34,34,34,42,42,54,34:
      REM  W
770  DATA 0,34,34,20,8,20,34,34: REM
      X
780  DATA 0,34,34,20,8,8,8,8: REM
      Y
790  DATA 0,62,32,16,8,4,2,62: REM
      Z
800  DATA 0,62,6,6,6,6,6,62: REM
      LEFT BRACKET
810  DATA 0,0,2,4,8,16,32,0: REM
      BACK SLASH
820  DATA 0,62,48,48,48,48,48,62:
      REM  PROMPT
830  DATA 0,0,8,20,34,0,0,0: REM
      ^
840  DATA 0,0,0,0,0,0,0,62: REM
     UNDERLINE
850  DATA 0,62,65,93,69,93,65,62:
                          ■
```


（請用英文填寫，方便電腦處理）　（只適用於本港訂戶）　信封面請註明「訂閱」

本人欲訂閱

□電腦時代一年（十二期）HK$120

□電腦時代＋CompuHouse磁碟一年（十二期）HK$475

由第＿＿期開始

姓名：＿＿＿＿＿＿＿＿

地址：＿＿＿＿＿＿＿＿

電話：＿＿＿＿＿＿＿＿

附上劃線支票 No. ＿＿＿＿＿＿＿＿

支票收款人請填寫 "Computing Age Publisher"

請寄九龍中央郵箱71193號

NO. ＿＿＿＿＿＿

WORDSTAR高級應用篇：

蘋果長老

WORDSTAR的運作備忘錄

WS（WORDSTAR）是MICROPRO INTERNATION－AL CORPORATION 在1979年前發展出來，用作將LINE PRINTER 當成電動打字機使用（經由WS程序控制），以協助使用者建立文件檔案，處理商業文件，亦可用來編輯及更改程式檔案，甚或印出磁碟上檔案，刪除磁碟上任何一檔案等，再回到CP／M 作業系去。事實上WS固然對擁有印字機的朋友帶來最佳文書處理效果，另一方面就算沒有印字機，WS仍可取代ED來產生有效的高階語言，滙編語言的原程式，以下我將WS的百多個命令簡化爲7個表，使各位在WS的應用上可以發揮最佳效果。

當各位開始WS時，從目錄中可以執行它的12個功能（表1），它們是由WS的12組內置副程式完成。

表(1)功能控制鍵

D	建立或編輯書信檔案
N	用來編輯或修改程式檔，數據檔
M	磁碟上必須存有MAILMRGE.OVR 檔案，這樣WS 會將列印控制權交予MAILMERGE處理,(有關MAILMERGE詳見第14，15期)
F	檔案目錄控制鍵
L	改變使用的磁碟機（由A改爲B，C，D等，若WS 在RAMDISK中運作時最好不要使用此項功能，因爲會令WS的大部份運作時間花在磁碟機上而降低速度）
R	執行磁碟上的其它命令檔案（假如WS是運作在名爲D：的RAMDISK時，此命令可將另一片磁碟上檔案以PIP D：＝？：XXXXX・XXX 形式迅速將該文檔抄錄至RAMDISK以D 命令直接編輯而不需離開WS
H	設定WS系統說明（HELD）的詳細程度
X	結束文字處理系統回到CP／M作業系統中
P	列印檔案
Y	刪除檔案
O	複製檔案
E	重定檔案名稱

編輯控制鍵(一)

當各位採用D或N來編輯檔案時，對CURSOR 及編輯功能的控制方面如下列所示（Λ代表CTRL 鍵）

ΛA	往左一個字
ΛS	往左一個字符
ΛD	往右一個字符
ΛF	往右一個字
ΛE	往上一行
ΛX	往下一行
ΛZ	CURSOR不動而螢幕往上捲動一行
ΛW	CURSOR不動而螢幕往下捲動一行
ΛO	從新設定螢幕格式
ΛC	CURSOR往下移動一螢幕長的距離
ΛR	CURSOR往上移動一螢幕長的距離
ΛDEL	CURSOR往左刪除一個字符，此時CURSOR右邊的字串亦往左移
ΛG	CURSOR往右刪除一個字符而CURSOR的位置不動
ΛT	CURSOR往右刪除一個字串而CURSOR的位置不動
ΛY	刪除一整行
ΛV	設定或取消插入功能
ΛI	移動一個欄位（TAB SET）
ΛP	報表之印製
ΛRET	結束段落
ΛU	取消指令
ΛN	插入一空白行
ΛB	把整個段落重新安排以符合邊界要求
ΛL	串列的尋找
ΛQ	命令的擴充
ΛJ	各種命令的說明
ΛK	段落移動或刪除或COPY

剛才ΛQ的擴充命令如下：

S	往左移到螢幕邊界，每次移動一個字符
E	往上移到螢幕頂，每次移動一行
X	往下移一行

WORDSTAR 的運作備忘錄

D　往右移一個字符
R　往上移到檔案頂端
C　往下移動到檔案底端
Z　CURSOR隨檔案往上捲動，直到按下ΛQ爲止。
W　CURSOR隨檔案往下捲動，直到按下ΛQ爲止
F　串列尋找，分成部分和整個串列的尋找
A　串列尋找後取替以另一些字組
Q　重覆命令直至按下SPACE BAR爲止

編輯控制鍵(二)

另一方面，當閣下使用D或N　命令編輯文檔之時，可以按入ΛQ連接以下的功能鍵

S　設定行間隔
L　設定左邊界
R　設定右邊界
I　設定欄位（SET TAB）
N　取消欄位
G　按入多次可以把整段書信或程式左邊界暫時定在n個欄位的位置
C　把CURSOR左邊的文字移至中央對齊
X　取消邊界
J　書信右邊對齊的控制鍵
F　對已鍵入的某一行文字作邊界及欄位的設定
SPACE　取消邊界

控制段落運作鍵

現在我們都已知道設定格式化是用以上的ΛO及所列鍵，而控制段落運作及貯存方面是要用ΛK及下列功能鍵的。

D　貯存經編輯過的檔案，然後回到非編輯狀態，以進行其它處理
X　更新檔案，然後離開文字處理系統，回到CP／M作業系統去
S　更新檔案，並回到原來離開的位置予使用者繼續編輯檔案
Q　不貯存更新的檔案，並回到非編輯狀態
B　段落設定起始點要先移動游標到該處，便會出現〔B〕〔K〕
K　段落設定終點要先移動游標到該處，便會出現〔B〕〔K〕
H　刪除〔B〕及〔K〕的段落標記
V　將已設定〔B〕及〔K〕標記的段落移到CURSOR所在處
C　將段落複製
Y　刪除段落
W　將該段落寫到磁碟去
R　從磁碟中讀一段落或檔案到某一檔案內
J　檔案刪除
O　檔案的複製
E　檔案名稱重定
P　檔案的印製
L　改變運作的磁碟機
F　檔案目錄開關控制鍵
0—9　設定和隱含記號0—9
SPACE　取消前置鍵 ■

LIST 程式變 BOOT 機

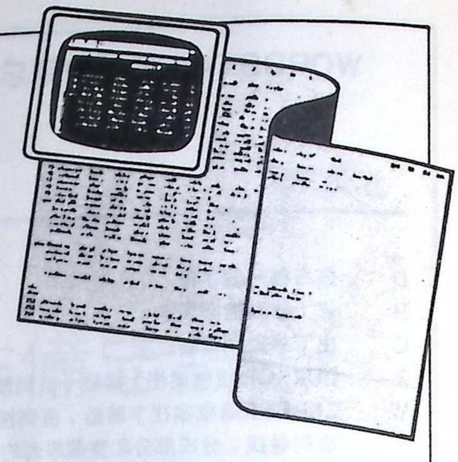
梁卓傑

今期爲大家介紹二種程式保護方法。第一種可說是非常有用的，就是輸入程式時，在任何一行中最後的一個格都可以容納一個DOS 指令。現在你照着下面按鍵，就會明白了：　按SPACE，〔按CTRL—D
BOOT一個DOS 3.3，按CTRL—RESET，打入NEW
打入1REMCOMPUTING—AGE—A—GOOD—BOOK—〔PR＃6
按LIST
「OH DEAR！嚇死人呀！」
因爲在電腦LIST到BOOK〔SPACE〕這時，便將後面的資料往下一行列出，所以在BOOK後面的SPACE便是最後一個格，以後的如果是CTRL—D DOS 指令便會辨行的。在BEAGLE BROS 的磁碟裏有一個叫BEAGLE NEWS 的程式，在LIST它的時候，怎麼會得一行字？再LIST的時候，程式也不見了！因爲它在最後的一個格之後加上了CTRL—D FP。FP 指令將程式洗掉，所以在第二次LIST中得出的結果就是.....

第二種方法是非常有趣的，依照下面的指示，就可以了
：1REM，按24次CTRL—J
按COMPUTING AGE IS GOOD
按10次CTRL—J，按輸入鍵〔RETURN〕
按LIST，怎樣？
這種方法可以和第一種結合，在LIST的時候，跳了幾十行，跟着便辨行後面的DOS 指令了，但位置要適當才可，大家都應該知道。■

CP/M 版本的VISICALC

SUPERCALC



陳耀南

攤頁分析表最著名的當然是VISICALC ，它是最早期設計出的軟件，用來預算銷售計劃，稅項，財務比率，個人預算，工程變化，成本估計及平衡支票簿上的數據往來，但可惜VISICALC不能在CP／M 的操作系統下運行，這對於擁有Z —80咭的用家似乎不能發揮其CPU的高速性能了。

SUPERCALC 是SORCIM公司在1981年推出的攤頁分析軟件，其實是CP ／M系統的VISICALC ，它們兩者之間有很多相同之處。在SUPERCALC 中數據有效位數能達到16位，文字長度在每一單元中可達116 個字符，而方程式或稱算式亦可長達116 個字符。同時SUPERCALC 的內置函數包括SUM，COUNT，AVERAGE，MIN，MAX，INT，ABS，SIN，COS，TAN，ASIN，ATAN，EXP，SQRT，LN，LOG 10，PI，IF，OR，AND，NOT，ERROR，NA，LOOKUP 及NPU等。

SUPERCALC 亦提供了准許不同長度的COLUMN 及同時提供保護單元功能，以避免錯誤地把重要的單元項清除或改變。

SUPERCALC 的運作指令和VISICALC 的亦大同小異，各位可利用本文作爲命令總表備忘，配合“？”命令取得解說螢幕而將厚厚的MANUAL 抛到一旁。

BLANK —清除未被保護的被指定單元爲空白。
COPY—把某部份的行，列或一方塊的單元翻印到被指定的單元位裏。
DELETE —清除一列，一行資料單元或一磁碟檔案。
EDIT —修改某指定單元的內容。
FORMAT —控制數值，文字及算式的顯示形式，不同的字符組合代表不同的意思和格式，其分別如下(GLOBAL)

- I —整數數值顯示。
- G—資料回復預置格式顯示。
- E—數值以指數格式顯示。
- S—數值以銀碼格式顯示。
- R—以右對齊形式顯示數值。
- L—以左對齊形式顯示數值。
- TR—以右對齊形式顯示文句。
- TL—以左對齊形式顯示文句。
- *—數值以BAR GRAPH圖形顯示。
- D—自動設定計算方向格式，先爲ROW再爲COLUMN。
- GLOBAL —它包含著下列的副命令。
- F—FORM OPTION，決定方程是CELL或VALUES。
- N—NEXT OPTION，決定顯示游標的亮起或關掉。
- B—BORDER OPTION，決定工作及決定行及列顯示數字清除與否，多在印字機輸出時使用。
- RC —決定依行或列次序來計算。
- M—以手動來控制SUPERCALC 重複計算。
- A—自動計算功能。
- INSERT—新的行或列插入。
- LOAD—從磁碟檔案載入SPREADSHEET。
- MOVE —移動整列或行到工作紙的另一位置。
- OUTPUT —將部份或整張工作紙輸出到印字機或磁碟上。
- DISPLAY —當時顯示的部份工作紙輸出
- CONTENTS —整張或部份工作紙輸出，這可分爲下列選擇：
  - PRINTER —印出到印字機上。
  - CONSOLE —在螢幕上顯示。

SUPERCALC

DISK 一將資料輸入磁碟檔案。

SETUP 一印出規格的設定，其置定選擇如下：

(a) L：設定每頁的行數

(b) W：設定每行的字數

(c) S：設定特別的CONTROL CODE輸出

(d) P：印出到印字機上

PROTECT一保護某資料單元CELL，RANGE或BLOCK。

QUIT 一SUPERSCALCS 終止，回到CP／M系統去。

REPLICATE一工作紙上資料的複製。

SAVE 一將工作紙資料全部存入磁碟檔案中。

TITLE 一將行或列固定在螢幕上：

H 一將當時的行及其上的行固定

V 一將當時的列及其所有左邊的列固定

B 一同時固定CURSOR所在位置的行及列

C 一取消固定功能

UNPROTECT一取消保護功能。

WINDOW 一將螢幕分成兩部份的工作紙：

H一水平分割

V 一垂直分割

C 一分割的清除

S 一同步回捲

U 一回捲異步

EXECUTE一SUPERCALC 可讓用者建立起命令檔案，執行（EXECUTE） 的功能是從磁碟中讀回命令檔案，然後依次執行檔案內的命令。

ZAP一清除記憶體中的電子工作紙之所有資料數據。■

捉蟲訓練—YES AND NO

今期的捉虫問題是讀者WONG KWOK LEUNG 寄來。錯處很簡單，讀者應可容易找到。問題如下：

當程序正確運行時，按入"Y"時，螢幕會出現「√」的圖案，而當按入"N"時將會出示「×」的圖案。但按N出現的圖案則是有問題，究竟應該怎辦。

```
10   TEXT : HOME
20   INPUT "ENTER (Y/N) ";A$
30   HGR : HCOLOR= 2
40   IF A$ = "Y" THEN 80
50   IF A$ = "N" THEN 70
60   GOTO 20
70   HPLOT 0,0 TO 279,159: HPLOT 2
      79,0 TO 0,159
80   HPLOT 279,0 TO 20,159 TO 0,11
      0
90   VTAB 21: INPUT "RUN/END(R/E):
      ";B$
100  IF B$ = "R" THEN  RUN
```

我們歡迎讀者提供〔捉蟲訓練〕問題。程式長度沒有限制，以越簡短越好。程式必須要以APPLESOFT 寫成，並符合下列各項要求：

1. 程式必須是APPLESOFT ，任何POKE或CALL機械語言的敍述必須正確。
2. 程式必須能出現某些結果，而不可出現SYNTAX ERROR。
3. 所有問題必須附加簡短說明及答案。

假如閣下的作品獲得刊登，我們將會送出十二期免費雜誌給你或將你的訂閱延長一年。

所有投稿請寄九龍中央郵政信箱 C. P.O BOX 71193 。信封面請寫上〔投稿捉蟲〕。■

答案：將行句70最後處加入「：GOTO」就能夠不使按入"N"時會同時顯示「×」和「√」的圖案。

16 K RAM咭改良之(二)

解決HANG 機的困擾

BABEL

前言

在APPLE Ⅱ的衆多介面咭之中，筆者對記憶體擴展咭獨有偏愛，尤其是16K RAM CARD，價錢廉宜，結構簡單，而用途却又如斯多樣化。所以筆者在改良這塊介面咭及爲它發掘更多用途方面，花了不少時間。在本刊第十二期中，筆者已經給各位介紹了如何增加狀態顯示器，而除了今期之外，估計在可見之未來，將會繼續發表其他關於16K RAM CARD的改良，對16K RAM CARD有濃厚興趣的讀者，請繼續留意。

閒話休提了，各位如果有16K RAM CARD的話，請嘗試依以下步驟鍵入，看有何反應？

開機後

］CALL－151〔CR〕

＊C083

怎樣？是不是你的APPLE 不再理會你的輸入了？連－CTRL－RESET 也毫無反應了？爲什麼？因爲你的16 K RAM CARD 在開機後，內裏的內容是隨機放置，毫無意義的。而＄C083 這一道指令，是把RAM CARD 的記憶替代了主機板上的＄D000－＄FFFF系統程式，如此當然會HA－NG 機的了，且連CTRL－RESET也完全不起作用。因爲RESET 的指位器是由6502 指定，放在＄FFFC及＄FFFD 的，而此刻RAM CARD 內的指位器的內容是亂數，故此全機完全失去控制了，你如要重新獲得控制權，只有關機再開機了。

你很好運，因爲你現在只是做實驗而已。如你在寫組合程序的過程中，不慎鍵入了如＄C083或＄C080這些指令，你將會失去所有在記憶體中的有用程式及資料。筆者就曾經在爲16K RAM CARD寫程序的時候，嘗試執行程式之後而失去控制，因而需是重新鍵入程式；老實說，就算不須再重新鍵入，但每嘗試執行一次，便要重新關機，BOOT 機一次，不是很煩嗎？當然，有些聰明的讀者，可能會提議先將APPLE 本身的 ＄F800 －＄FFFF 系統程式搬入 RAM CARD，不是什麼問顏也解決了？但如果你所寫的程式，有如RAM DISK 程式般，是要完全使用RAM CARD 所有記憶空間的，那又如何呢？還有，某些程式在執行後，遇到外界RESET的話，是會利用RAM CARD 來HANG 機的（如果RAM CARD 上有狀態顯示器的話，讀者是可以看到的），此時便可以利用這小小的改良，來脫離RAM CARD 的困鎖了。總之，就筆者經驗所得，這雖然只是一個很小的改良，却幫了筆者不少大忙。

原理概述

筆者在本刊第十二期中，經已介紹過，幾乎市面上的16 K RAM CARD 中，都有一隻74LS175 的TTL ，它是負責選擇RAM CARD 不同狀態的，它在RAM CARD 中的綫路部份可參閱FIG 1，換句話說，它是16K RAM CARD 的心臟部份，只要花點心思，便可以利用它作出一些改良，例如：如何加上狀態顯示器便是了。這隻IC內是 4 隻 D 型觸發器，而觸發器（FLIP－FLOP) 是有一種特性，就是當它的

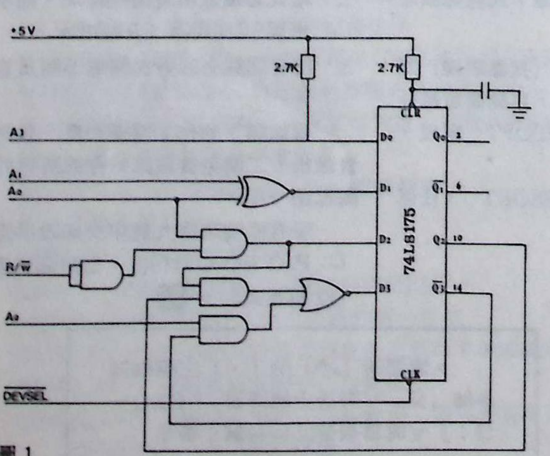


圖 1

▽－H ▲－L	Q0	$\overline{Q0}$	Q1	$\overline{Q1}$	Q3	$\overline{Q3}$
C080	▲	▽	▽	▲	▽	▲
C081	▲	▽	▲	▽	▲	▽
C082	▲	▽	▲	▽	▽	▲
C083	▲	▽	▽	▲	▲	▽
C088	▽	▲	▽	▲	▽	▲
C089	▽	▲	▲	▽	▲	▽
C08A	▽	▲	▲	▽	▽	▲
C08B	▽	▲	△	▲	▲	▽

圖 2

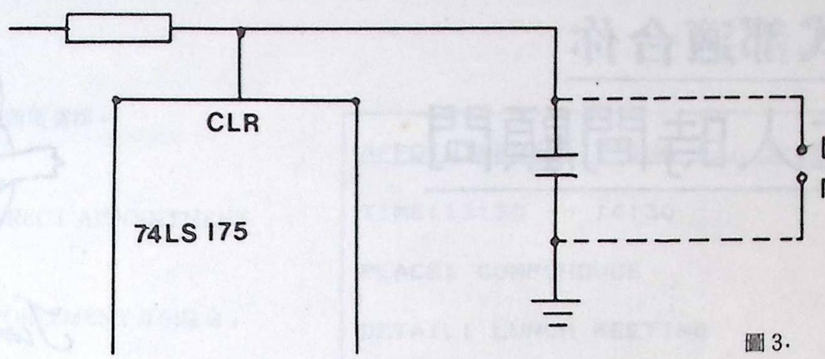


圖 3.

CLR脚，輸入"LO"的話，它的輸出便立即變爲"LO"，換句話說，當我們在74LS175的CLR脚輸入"LO"的話，它的四個輸出——Q0，Q1，Q2及Q3，便會立即變爲"LO"了，由FIG 2 的表中可見，當Q0，Q1及Q3均爲"LO"時，RAM CARD 的對應狀態是在＄C081—READ ROM & WRITE RAM 。現在各位讀者應該明白，無論16 K RAM CARD 目前處於什麽狀態（包括被HANG），只要在74LS175的CLR 脚輸入"LO"，RAM CARD 便立即變爲＄C081的狀態。但究竟怎樣，才可以輸入"LO" 到它的CLR 脚呢？在FIG 1中，可看到74LS175 的CLR脚與—GND 之間是連接有一個電容的，所以我們只需要接上一個小型的按鈕（常開開關），在這個電容的兩端，如FIG 3所示；在平時這個開關是開路的，所以16K RAM CARD 照常工作，當有需要時，一按這按鈕，RAM CARD 便立即跳回＄C081 的狀態。

後記

這次的改良，雖然是十分微型，但很實用，希望各位能夠領略到，更希望對16K RAM CARD 有興趣的朋友，想想看還有什麽地方值得改良，或發掘更多利用RAM CARD 的途徑，寄來公開發表，讓大家一起分享。■

各位業餘程序編寫員萬勿錯過本文

電腦時代爲你開闢一條新途徑

各位業餘程序編寫員，你們的軟件創作才華有沒有被人忽視？你們的心血作品有沒有人賞識？

我們知道，你們這群業餘程序編寫員絕大部份都是天才橫溢，所創作編寫的軟件並不遜色於任何的職業程序設計師，而且，有許多更是超乎市面要求的商業水準。你們的作品之中，有許多是經年累月的經驗。我們懇請你們不要埋沒自己的天才，將你的經驗創作成果與每一個電腦迷分享。

電腦時代出版社與 COMPUHOUSE 軟件製作所願意爲你全盤策劃出版你的軟件創作成果，讓它流傳到每個電腦用家手中。你絕對不用担心任何的問題，我們會代你効勞一切有關版權、出版、發行製作等問題，你只要安心在家創作便可。

我們擁有豐富經驗的行政人員、軟件工程師、編採出版人材。而且更具有一個廣大的市場發行網，我們的發行網除遍達香港外，還推廣到台灣、星馬泰等東南亞國家，而且更延伸至美國及加拿大。中國大陸這個龐大的市場當然也在我們足跡之下。

透過我們出版軟件，你將會得到最大的收穫。

我們龐大隊伍的生產綫，不單設備完善，且有足夠能力應付任何各方面的需求。

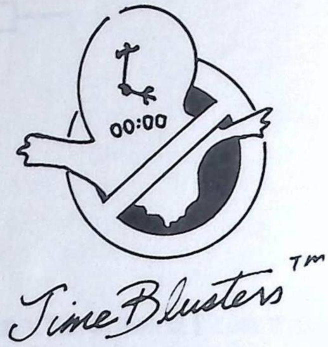
我們歡迎任何機種及類型的軟件創作，無論是IBM、APPLE電腦、由娛樂軟件、教育應用至商業軟件，只要你們有信心認爲自己的作品有一定的商業水準，我們都樂意爲你出版。

歡迎你們携同作品前來本出版社商談，或電 3—312007 與譚先生聯絡。

無論你是正牌大忙人或是見習大忙人，

本程式都適合你

大忙人時間顧問



黃博思

你會否爲每日繁重的約會或未能善用和盡用分秒而懊惱？又或者發覺計算和找尋空檔來應付約會是件麻煩費時的事？又或，記事簿不足夠紀錄你每天的約會？有了「大忙人顧問」，以上都成爲歷史；它是個替你爭取分秒，助你訂定約會，記錄約會的程式。快把它INIT 在你的磁碟上，每天印份日程表帶在身，以後就讓它替你處理約會和助你制訂日程表吧！

操作準備

本程式能在一台或以上磁碟機的蘋果系統上操作。爲了針對不同人士的需求，本程式有多處是彈性及方便地讓你修改的：

1：LINE 90的PS是印字機SLOT NO ，同行的CP＄等是ESPON 印字機的字體選擇，目前用正常字體。用其他印字機便只須修改此行。

2：LINE 100 的H＝1把週日定爲假期，不接受約會，若你整週都受約，可把H＝1改爲H＝0，又H＝2即週一爲假期，由此類推。

3：LINO 110 的DN和DS分別是PRINE NO. 和SLOT，雖然在操作中亦可改變DN和DS 。但若你的DRIVE 常不在SLOT 6的話，改DS 可免每次RUN都改OPTION。

4：LINE 130 的DIMA＄（100）…等三個100是約會最大項數的兩倍，即目前可每日有50個約會。

5：LINE 200 的IF述句中，把M＄(1)和H＄(1) 定爲零，H＄(2)定爲24，M＄(2)定爲零，即「接受約會時段」是00：00——24：00（程式採用國際時間制，即一天廿四小時也可約會），你應按需要改ARRAY 1及2的值。

6：最後， LINE 455 把CTRL－C 鎖住，你只要刪去本行便可用CTRL－C BREAK 程式了。但你便不能隨意返回MENU了，注意！

操作

程式的操作在螢幕上有清楚指示，大部份輸入都需證實及允許修改的，輸入時間需用數字塡入劃定的時分格中，注意約會時間不可早過或遲過你的「接受約會時段的」，在RUN時首先你會看到如圖(一)的主頁：

```
DISK OPTION -   SLOT:6  NO:1

CURRENT DATE OF MEMO:15/ 3/85  FRI

NO. OF BOOKING 1

           WORKING MENU
-----------------------------------------

          DIRECT ENTER APPOINTMENT
          CHECK/ENTER APPOINTMENT
          EDIT APPOINTMENT
          CANCEL APPOINTMENT
          LIST APPOINTMENT
          CHANGE CURRENT DATE
          CATALOG/DELETE FILES
          CHANGE DISK OPTIONS
          QUIT

<=ARROWS SELECT=>..[RETURN] EXECUTE
```

圖(1)

主頁顯示了主菜單，磁碟系統，約會日期和項數等，另外，錯誤訊息和操作指示亦會在主頁底部顯示，選項目方法是把反白方框移到欲報行的項目上後按RETURN。 但在報行各項前，必須輸入欲訂約會的日期。（INIT CURRENT DATE）

直接輸入約會

這部份適合一些你約會人或處理一些必要做但不太限時間的事項。因爲你須輸入該事項預計所需時間讓「大忙人顧問」替你找出空檔；若在當日找不到空檔，便自動讀下一日，直至找到空檔爲止。若你要處理一些有時間性但無需一定做的事項，或不想「顧問」替你擇日，下一個功能便幫到你。

檢索/輸入約會

這部份中，你先輸入約會時間，「顧問」替你找出當時你是否空閒，若空閒便印出上約會結束時間和下約會開始時間，與及地點細節來讓你預計交通時間等來訂約。若該時段

大忙人時間顧問

已被預約，便印出上下兩個空檔及四項選擇。

1：接受提供的空檔

2：另選一日，保持約會時間不變

3：在同日選其他時間，即跳到DIRECT APPOINTMENT去

4：放棄約會，回主菜單

相信DIRECT及CHECK APPOINTMENT互相採合，定能適合各種預訂的形式的。

編改約會

若你欲修改地點和細節，又或欲在同一時間內接受另一個約會，可用EDIT在細節上補充。

取消約會

取消約會，在取消前會先要你證實。

列出約會

你可選用印字機或監視器輸出，用後者可按SPACE BAR 作快速列印，即「飛」去不看的約會。前者可用多種字體印刷（見操作準備）。

改變即時日子

讓你選另一日期頁，「顧問」替你算出週日。若你想增加休息日，可改LINE 1240的IF述句。CATALOG／DEL－ETE FILES，CHANG DISK OPTIONS和QUIT 等都非常簡單，只要照指示做便成。這裏不再詳述。

程式原理

程式運作主要是ARRAY 的使用，由於在國際時間制中，較遲的時間的值比較早的時間值大的。（時間本就是向量 VECTOR）把ARRAY 順序排列，便得到下列一個「可計算」的ARRAY表：

時 間	時 ： 分
約會 1	H $(0)：M $(0)，A $(0)地點
	H $(1)：M $(1)，A $(1)細節
約會 2	H $(2)：M $(2)，A $(2)地點
	H $(3)：M $(3)，A $(3)細節
⋮	⋮ ⋮
約會 N	

從表中，可見空檔是下約會開始時間減上約會的結束時間，即VAL H $(2)＊60＋M 8 (2)—VAL H($)(2)— 〔H $(1)＊60＋M $(1)〕，把每段空檔和輸入的空檔（DIRECT EN－TER APPOINTMENT中），比較若大過或相等便是有空了。

```
APPOINTMENT LISTING 16/ 3/85  SAT

TIME:13:30 -- 14:30

PLACE: COMPUHOUSE

DETAIL: LUNCH MEETING
```

日程表HARDCOPY輸出例子。

另外，H $(0)：M $(0)和H $(1)：M $(1)之間的時間是預訂了的，在CHECK APPOINTMENT時，若輸入的時間夾在中間，便表示不能在當時預約了。

若讀入的日子檔是空的，則END OF DATA 必然發生，故用ON ERR GOTO破解。

其他功能如LIST／CANCEL／EDIT／等都是一般的ARRAY 運算，相信你也遇過不少同樣的例子，這裏不再詳述。值得一提是程式的顯示和輸入形式，可在寫其他實用程式時參考。

最後補充：時分ARRAY 用 STRING 的原因只是爲了方便輸入和顯示，和計算60進位的時間。

本文程式長度爲36 SECTOR ，對於大忙人來說可能沒有時間打入，所以我們將本程式收錄在本期磁碟上，大家可以在磁碟上找TIME BLUSTERS這個FILE，亦即這個大忙人顧問的程式了。■

```
10  REM  *************************

20  REM  * THE BUSYMAN'S FRIEND *

30  REM  * "TIME BLUSTER (TM)"  *

40  REM  *   CREATED 84-85 BY   *

50  REM  * BUSYGUY-WONG POK SZE *

60  REM  *************************

70  ONERR  GOTO 430
80  REM  EPSON PRINTER SETTING
90  CLEAR :PS = 1:CP$ =  CHR$ (18
    ):UL$ =  CHR$ (27) + "-":US$
     =  CHR$ (1):UN$ =  CHR$ (0)
    : REM  SLOT 1,NORMAL CHARACT
    ER:UL$=UNDERLINE
100 DA$ = "SUNMONTUEWEDTHUFRISAT"
    :H = 1: REM  SUNDAY = HOLIDA
    Y
```

```
110 DN = 1:DS = 6: REM  DEFAULT D
    RIVE NO & SLOT
120  REM  DEF. VARIABLES
130  TEXT : DIM A$(100),M$(100),H
     $(100): REM  1/2 MAX DATING
140 B$ =  CHR$ (7) +  CHR$ (7):D$
     =  CHR$ (13) +  CHR$ (4): FOR
     I = 1 TO 30:SP$ = SP$ + " ":
      NEXT
150  REM  OPTIONS
160  FOR I = 1 TO 9: READ B$(I):B
     $(I) =  LEFT$ (SP$,10) + B$(
     I) +  LEFT$ (SP$,30 -  LEN (
     B$(I))): NEXT : FOR I = 1 TO
     4: READ B1$(I):B1$(I) =  LEFT$
     (SP$,10) + B1$(I) +  LEFT$ (
     SP$,30 -  LEN (B1$(I))): NEXT
170  DATA "DIRECT ENTER APPOINTME
     NT","CHECK/ENTER APPOINTMENT
     ","EDIT APPOINTMENT", "CANCE
     L APPOINTMENT","LIST APPOINT
     MENT","CHANGE CURRENT DATE",
     "CATALOG/DELETE FILES","CHAN
     GE DISK OPTIONS","QUIT"
180  DATA "ACCEPT ALTERNATIVES","
     SAME TIME..ANOTHER DAY","SAM
     E DAY..ANOTHER TIME","MENU"
190  REM  SET WORKING HOUR
200  IF N = 0 THEN N = 3:M$(1) =
     "00":H$(1) = "00":M$(2) = "0
     0":H$(2) = "24":M$(3) = M$(2
     ):H$(3) = H$(2)
210  REM  WORKING MENU
220  POKE 34,0: HOME : INVERSE : PRINT
     " TIME BLUSTER TIME BLUSTER
     TIME BLUSTER ": NORMAL
230  VTAB 3: HTAB 2: PRINT "DISK
     OPTION -  SLOT:";: INVERSE :
      PRINT DS;: NORMAL : HTAB 25
     : PRINT "NO:";: INVERSE : PRINT
     DN: NORMAL
240  VTAB 5: HTAB 2: PRINT "CURRE
     NT DATE OF MEMO:";: INVERSE
     : PRINT DT$;"  ";WK$;: NORMAL
     : HTAB 25: PRINT "/";: HTAB
     28: PRINT "/";: HTAB 31: PRINT
     " "
250  VTAB 7: HTAB 2: PRINT "NO. O
     F BOOKING ";: INVERSE : PRINT
     (N - 3) / 2: NORMAL
260  HTAB 13: VTAB 9: PRINT "WORK
     ING MENU": FOR I = 1 TO 40: PRINT
     "-";: NEXT
270  VTAB 12: FOR I = 1 TO 9: PRINT
     B$(I);: NEXT :V = 1: VTAB 22
     : HTAB 2: PRINT "<=ARROWS SE
     LECT=>..[RETURN] EXECUTE"
280  REM  CHECK DATE LINE
290  IF DT$ = "" THEN  VTAB 17: INVERSE :
     PRINT B$(6): NORMAL : HTAB 2: VTAB
     22: PRINT B$;"INIT CURRENT D
     ATE OF MEMO"; TAB( 39): GOSUB
     1140: GOTO 200: REM  INIT DA
     TE
300 K =  PEEK ( - 16384): VTAB 11
     + V: INVERSE : PRINT B$(V):
     NORMAL : IF K < 128 THEN 30
     0
310  POKE  - 16368,0: IF K = 141 THEN
     ON V GOSUB 550,670,920,840,
     1010,1140,1410,360,1480: GOTO
     200
320  IF K = 149 THEN V = V + 1: VTAB
     10 + V: NORMAL : PRINT B$(V -
     1): IF V > 9 THEN V = 1: VTAB
     20: NORMAL : PRINT B$(9): REM
      IF END OF OPTION,RESTORE CU
     SOR
330  IF K = 136 THEN V = V - 1: VTAB
     12 + V: NORMAL : PRINT B$(V +
     1): IF V < 1 THEN V = 9: VTAB
     20: INVERSE : PRINT B$(9): VTAB
     12: NORMAL : PRINT B$(1): REM
      IF END OF OPTION,RESTORE CU
     SOR
340  GOTO 300: REM  GET KEY AGAIN
350  REM
360  REM  CHANGE CONFIG.
370  INVERSE : HTAB 22: VTAB 3: GET
     A$: IF  ASC (A$) > 48 THEN D
     S =  VAL (A$)
380  HTAB 22: PRINT DS;: HTAB 28:
      GET A$: IF  ASC (A$) > 48 THEN
     DN =  VAL (A$)
390  HTAB 28: PRINT DN: IF DN > 2
      OR DS > 6 THEN  PRINT B$: GOTO
     370
400  NORMAL : GOTO 220
410  REM
420  REM  ERROR HANDLING
430 EL =  PEEK (218) +  PEEK (219
     ) * 256:ER =  PEEK (222): PRINT
     D$;"PR#0"
440  IF ER = 255 THEN  PRINT B$: GOTO
     200
450  IF ER = 5 THEN  VTAB 22: PRINT
     B$;"  NO APPOINTMENT ON ";: INVERSE :
     PRINT DT$;: NORMAL : PRINT  TAB(
     39): FOR P = 1 TO 1000: NEXT
     : GOTO 200
460  IF ER = 22 THEN 200
470  IF ER = 4 THEN  HTAB 1: VTAB
     22: PRINT B$;"  DISK WRITE P
     ROTECTED"; TAB( 39): FOR P =
     1 TO 1000: NEXT : GOTO 200
480  IF ER = 6 THEN  HTAB 1: VTAB
     22: PRINT B$;"  FILE NOT FOU
     ND"; TAB( 39): FOR P = 1 TO
     1000: NEXT : GOTO 200
490  IF ER = 254 THEN  HTAB 1: VTAB
     22: PRINT B$;"  ILLEGAL INPU
     T"; TAB( 39): FOR P = 1 TO 1
     000: NEXT : GOTO 1150
```

```
500  IF ER = 9 THEN  HTAB 1: VTAB
     22: PRINT B$;"   DISK FULL,SA
     VE CANCELLED"; TAB( 39): PRINT
     D$;"DELETEAPP:";DT$: GOTO 20
     0
510  IF ER = 8 THEN  VTAB 22: HTAB
     1: PRINT B$;"   I/O ERROR"; TAB(
     39): FOR P = 1 TO 1000: NEXT
     : GOTO 200
520  TEXT : HOME : NORMAL : PRINT
     B$;"ERROR CODE ";ER;" AT LIN
     E ";EL: PRINT "CONSULT YOUR
     DOS OR APPLOESOFT MANUAL": END
530  REM
540  REM  DIRECT ENTER APPOINTMEN
     T
550  POKE 34,0: HOME :S = 0: INVERSE
     : PRINT "DIRECT ENTER APPOIN
     TMENT"; TAB( 41): NORMAL : POKE
     34,2: VTAB 3: PRINT "ENTER I
     NTERVAL ";:Z = 16: GOSUB 150
     0
560  PRINT : PRINT : PRINT "DATE:
     ";DT$;"   ";WK$: PRINT : PRINT
     "YOU MAY BE FREE AT:-": PRINT
570  FOR I = 1 TO N STEP 2: IF ((
      VAL (H$(I + 1)) * 60 +  VAL
     (M$(I + 1))) - ( VAL (H$(I))
      * 60 +  VAL (M$(I)))) >  =
     ( VAL (H$) * 60 +  VAL (M$))
      THEN S = 1: HTAB 2: PRINT H
     $(I);":";M$(I);" -- ";H$(I +
     1);":";M$(I + 1): REM  COMPU
     TE FREE TIME
580  NEXT : IF S = 0 THEN  PRINT
     "NO TIME,CHOSE ANOTHER DATE"
     : GOSUB 1560: GOSUB 1310: GOTO
     570
590  PRINT : PRINT : PRINT " KEY
     ";: INVERSE : PRINT "A";: NORMAL
     : PRINT " FOR APPOINTMENT..A
     NYKEY TO MENU";: GET A$: PRINT
     : IF A$ <  > "A" THEN 200
600  PRINT : PRINT "START TIME: "
     ;:Z = 13: GOSUB 1500:H$(N +
     1) = H$:M$(N + 1) = M$: PRINT
     : PRINT
610  HTAB 3: PRINT "END TIME: ";:
     Z = 13: GOSUB 1500:H$(N + 2)
      = H$:M$(N + 2) = M$: PRINT
     : PRINT : INPUT " PLACE:";A$
     (N + 1): PRINT : INPUT "DETA
     IL:";A$(N + 2):N = N + 2
620  FOR I = 0 TO N - 2 STEP 2: FOR
     J = 0 TO N - 2 STEP 2: IF  VAL
     (H$(J) + M$(J)) <  =  VAL (H
     $(J + 2) + M$(J + 2)) THEN 6
     40: REM  ARRANGE APPOINTMENT
630 H1$ = H$(J):M1$ = M$(J):H2$ =
     H$(J + 1):M2$ = M$(J + 1):H$
     (J) = H$(J + 2):M$(J) = M$(J
      + 2):H$(J + 1) = H$(J + 3):
     M$(J + 1) = M$(J + 3):M$(J +
     2) = M1$:H$(J + 2) = H1$:M$(
     J + 3) = M2$:H$(J + 3) = H2$
     :A1$ = A$(J):A2$ = A$(J + 1)
     :A$(J) = A$(J + 2):A$(J + 1)
      = A$(J + 3):A$(J + 2) = A1$
     :A$(J + 3) = A2$
640  NEXT J,I: PRINT : PRINT "APP
     OINTMENT ARRANGED": FOR P =
     1 TO 1000: NEXT : GOTO 200
650  REM
660  REM  CHECK/ENTER APPOINTMENT
670  HOME : INVERSE : PRINT "CHEC
     K/ENTER APPOINTMENT"; TAB( 4
     1): NORMAL : VTAB 3: PRINT "
     ENTER START TIME ";:Z = 18: GOSUB
     1500: PRINT :TI =  VAL (H$ +
     M$)
680 J = 1: FOR I = 2 TO N - 2 STEP
     2: IF  VAL (H$(I) + M$(I)) <
     = TI THEN J = J + 1: IF TI <
     =  VAL (H$(I + 1) + M$(I +
     1)) THEN 740: REM  GOT IT !!
690  NEXT :J = J * 2: POKE 34,4: HOME
     : PRINT "YOU ARE FREE AT ";H
     $;":";M$: PRINT : IF J < 3 THEN
     PRINT "NO PRIOR APPOINTMENT
     ": PRINT : PRINT "YOU ARE FR
     EE FROM ";H$(1);":";M$(1): PRINT
     : GOTO 710
700  PRINT "PRIOR APPOINTMENT END
      AT ";H$(J - 1);":";M$(J - 1
     ): PRINT : PRINT "PLACE:";A$
     (J - 2): PRINT : PRINT "DETA
     IL:";A$(J - 1): PRINT
710  IF J >  = N - 1 THEN  PRINT
     "NO APPOINTMENT FOLLOWS": PRINT
     : PRINT "YOU ARE FREE TO ";H
     $(3);":";M$(3): GOTO 590
720  PRINT : PRINT "NEXT APPOINTM
     ENT START AT ";H$(J);":";M$(
     J): PRINT : PRINT "PLACE:";A
     $(J): PRINT : PRINT "DETAIL:
     ";A$(J + 1): PRINT
730  GOTO 590
740  POKE 34,2: HOME : VTAB 3: PRINT
     H$;":";M$;" WAS OCCUPIED ": VTAB
     7: PRINT "ALTERNATIVES - ";H
     $(I - 1);":";M$(I - 1);" --
     ";H$(I);":";M$(I): VTAB 8: HTAB
     11: PRINT "OR.. ";H$(I + 1);
     ":";M$(I + 1);" -- ";H$(I +
     2);":";M$(I + 2)
750  VTAB 13:V = 1: FOR I = 1 TO
     4: PRINT B1$(I);: NEXT : VTAB
     19: HTAB 2: PRINT "<=ARROWS
     SELECT=>..[RETURN] EXECUTE"
760 K =  PEEK ( - 16384): VTAB 12
      + V: INVERSE : PRINT B1$(V)
     : NORMAL : IF K < 128 THEN 7
     60
770  POKE  - 16368,0: IF K = 141 THEN
      ON V GOSUB 810,1140,550,200
     : GOTO 670
```

```
780  IF K = 149 THEN V = V + 1: VTAB
     11 + V: NORMAL : PRINT B1$(V
      - 1): IF V > 4 THEN V = 1: VTAB
     16: NORMAL : PRINT B1$(4): REM
     IF END OF OPTION,RESTORE CU
     SOR
790  IF K = 136 THEN V = V - 1: VTAB
     13 + V: NORMAL : PRINT B1$(V
      + 1): IF V < 1 THEN V = 4: VTAB
     16: INVERSE : PRINT B1$(4): VTAB
     13: NORMAL : PRINT B1$(1): REM
      IF END OF OPTION,RESTORE CU
     SOR
800  GOTO 760: REM  GET KEY AGAIN
810  POKE 34,13: HOME : GOTO 600
820  REM
830  REM  CANCEL APPOINTMENT
840  HOME : INVERSE : PRINT "CANC
     EL APPOINTMENT"; TAB( 41): NORMAL
     : VTAB 3: PRINT "ENTER START
      TIME ";:Z = 18: GOSUB 1500:
      PRINT : FOR I = 2 TO N - 2:
      IF H$(I) + M$(I) = H$ + M$ THEN
     870: REM  GOT IT !!
850  NEXT : VTAB 22: HTAB 5: PRINT
     B$;"NO APPOINTMENT AT THAT T
     IME": FOR P = 1 TO 1000: NEXT
     : GOTO 200
860  REM  CANCEL
870  PRINT : PRINT "THE APPOINTME
     NT TO BE ";: INVERSE : PRINT
     "CANCEL";: NORMAL : PRINT "
     IS:-": VTAB 7: PRINT "TIME:"
     ;H$(I);":";M$(I);" -- ";H$(I
      + 1);":";M$(I + 1): PRINT :
      PRINT "PLACE: "A$(I): PRINT
     : PRINT "DETAIL: ";A$(I + 1)
     : PRINT
880  PRINT "HIT ";: INVERSE : PRINT
     " Y ";: NORMAL : PRINT " TO
     CONTIRM ";: GET A$: IF A$ <
      > "Y" THEN 200
890  FOR J = I TO N:H$(J) = H$(J +
     2):M$(J) = M$(J + 2):A$(J) =
     A$(J + 2): NEXT :N = N - 2: HTAB
     3: VTAB 22: PRINT "THE APPOI
     NTMENT IS CANCELLED"; TAB( 3
     9): FOR P = 1 TO 1000: NEXT
     : GOTO 200
900  REM
910  REM  EDIT APPOINTMENT
920  HOME : INVERSE : PRINT "EDIT
      APPOINTMENT "; TAB( 41): NORMAL
     : VTAB 3: PRINT "ENTER START
      TIME ";:Z = 18: GOSUB 1500:
      PRINT : FOR I = 2 TO N - 2 STEP
     2: IF H$(I) + M$(I) = H$ + M
     $ THEN 950: REM  GOT IT !!
930  NEXT : VTAB 22: HTAB 5: PRINT
     B$;"NO APPOINTMENT AT THAT T
     IME": FOR P = 1 TO 1000: NEXT
     : GOTO 200
940  REM  EDIT
950  VTAB 5: PRINT "THE APPOINTME
     NT TO BE ";: INVERSE : PRINT
     "EDIT";: NORMAL : PRINT " IS
     :-": VTAB 7: PRINT "TIME:";H
     $(I);":";M$(I);" -- ";H$(I +
     1);":";M$(I + 1): VTAB 9: PRINT
     "PLACE: ";A$(I):V =  PEEK (3
     7) + 2: VTAB V: PRINT "DETAI
     L: ";A$(I + 1)
960  VTAB 22: PRINT "  [RETURN]=R
     ESERVE OLD VALUE..": VTAB 9:
      HTAB 7: INPUT " ";A$: IF A$
      <  > "" THEN A$(I) = A$
970  VTAB 9: HTAB 8: PRINT A$(I):
      VTAB V: HTAB 8: INPUT " ";A
     $: IF A$ <  > "" THEN A$(I +
     1) = A$
980  VTAB 11: HTAB 9: PRINT A$(I +
     1): VTAB 22: PRINT "  EDIT E
     XECUTED"; TAB( 39): FOR P =
     1 TO 1000: NEXT : GOTO 220
990  REM
1000  REM  LIST APPOINTMENT
1010  HOME : INVERSE : PRINT "LIS
     T APPOINTMENT"; TAB( 41): NORMAL
     : IF N = 3 THEN  VTAB 22: HTAB
     3: PRINT "NO APPOINTMENT IN
     MEMORY": FOR P = 1 TO 1000: NEXT
     : GOTO 200
1020  VTAB 3: HTAB 1: PRINT "LIST
      TO PRINTER": VTAB 5: HTAB 6
     : PRINT "OR MONITOR (P/M) :"
     ;: GET A$: IF A$ = "P" THEN
     1100
1030  IF A$ <  > "M" THEN 1020
1040  POKE 34,2: HOME : PRINT "AP
     POINTMENT LIST:";: INVERSE :
      PRINT DT$;"  ";WK$;: NORMAL
     : HTAB 26: PRINT "  ": POKE
     34,4: VTAB 22: PRINT "  [SPA
     CE]=FAST FORDWARD.."; TAB( 3
     9): POKE 35,21
1050  FOR I = 2 TO N - 2 STEP 2:P
      = 0: HOME : PRINT "TIME:";H
     $(I);":";M$(I);" -- ";H$(I +
     1);":";M$(I + 1): PRINT : PRINT
     "PLACE: ";A$(I): PRINT : PRINT
     "DETAIL: ";A$(I + 1): PRINT
1060 K =  PEEK ( - 16384): IF K =
     160 THEN  POKE  - 16368,0: GOTO
     1090
1070 P = P + 1: IF P > 70 THEN 10
     90
1080  GOTO 1060
1090  NEXT : POKE 35,24: PRINT : PR#
     0: HTAB 3: VTAB 22: PRINT "T
     HAT'S ALL OF THEM,ANY KEY TO
      MENU ";: GET A$: PRINT : GOTO
     200
1100  PRINT D$;"PR#";PS: PRINT CP
     $: PRINT UL$ + US$;"APPOINTM
     ENT LISTING ";DT$;"  ";WK$: PRINT
     UL$ + UN$: FOR I = 2 TO N -
     2 STEP 2: PRINT "TIME:";H$(I
     );":";M$(I);" -- ";H$(I + 1)
```

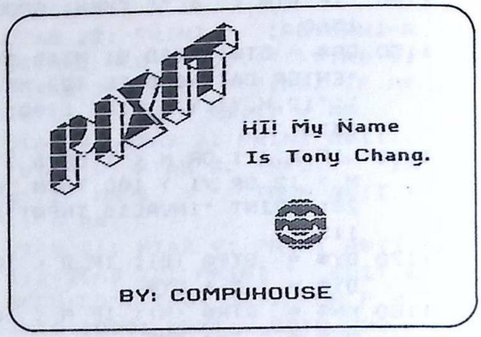
```
      ;":";M$(I + 1): PRINT : PRINT
      "PLACE: ";A$(I): PRINT : PRINT
      "DETAIL: ";A$(I + 1): PRINT
      : NEXT
1110  PRINT D$;"PR#0": GOTO 200
1120  REM
1130  REM  CHANGE DATE
1140  IF DT$ <  > "" THEN  GOSUB
      1560
1150 DO$ = DT$: VTAB 5: HTAB 2: INPUT
      "ENTER DATING DATE (DD,MM,YY
      ):";D,M,Y1:Y = Y1 + 1900: GOSUB
      1330
1160  IF D < 1 OR M < 1 OR D > E OR
      M > 12 OR Y1 > 100 THEN  HTAB
      28: PRINT "INVALID INPUT": GOTO
      1150
1170 DY$ =  STR$ (D): IF D < 10 THEN
      DY$ = " " + DY$
1180 MN$ =  STR$ (M): IF M < 10 THEN
      MN$ = " " + MN$
1190 YR$ =  STR$ (Y1):DT$ = DY$ +
      "/" + MN$ + "/" + YR$
1200  REM  MOVE DAY FORWARD
1210  GOSUB 1380:Z = C
1220  IF Y < 1701 THEN Y = 2099
1230  IF Y > 2099 THEN Y = 1701
1240  GOSUB 1380:N = C - Z: IF (M
      < 3 AND Y = 1900) OR Y < 19
      00 THEN C = C + 1:N = N + 1
1250  IF (M < 3 AND Y = 1800) OR
      Y < 1800 THEN C = C + 1:N =
      N + 1
1260 C = C - 7 *  INT (C / 7) + 1
      :WK$ =  MID$ (DA$,3 * C - 2,
      3): IF C <  > H THEN 1300: REM
       NOT HOLIDAY
1270  VTAB 22: PRINT B$;"IT'S ";:
      INVERSE : PRINT WK$;: NORMAL
      : PRINT ".[A]=ACCEPT DAY.[F]
      =NEXT DAY";: GET A$: IF A$ =
      "F" THEN 1310
1280  IF A$ = "A" THEN 1300
1290  GOTO 1270
1300  GOSUB 1540: RETURN
1310 D =  VAL (DY$):M =  VAL (MN$
      ):Y =  VAL (YR$) + 1900:D =
      D + 1: IF D > E THEN M = M +
      1: IF M > 12 THEN Y = Y + 1
1320  GOSUB 1330: GOTO 1170
1330 E = 31: IF M = 2 THEN E = 28

1340  IF M = 4 OR M = 6 OR M = 9 OR
      M = 11 THEN E = 30
1350  IF M = 2 AND Y / 4 =  INT (
      Y / 4) THEN E = 29
1360  IF M = 2 AND (Y = 1800 OR Y
      = 1900) THEN E = 28
1370  RETURN
1380 A = M + 13 - 12 * (M > 2):B =
      Y - 1 + (M > 2):C =  INT (36
      5.25 * B) +  INT (30.6 * A) +
      D - 621049: RETURN
1390  REM
1400  REM  CATALOG/DELETE FILES
1410  HOME : INVERSE : PRINT "CAT
      ALOG/DELETE FILES"; TAB( 41)
      : NORMAL : POKE 34,2
1420  VTAB 3: HTAB 1: PRINT "CATA
      LOG TO PRINTER": VTAB 5: HTAB
      9: PRINT "OR MONITOR (P/M) :
      ";: GET A$: IF A$ = "P" THEN
       PRINT D$;"PR#";PS: GOTO 144
      0
1430  IF A$ <  > "M" THEN 1420
1440  PRINT D$;"CATALOG";",S";DS;
      ",D";DN: PRINT D$;"PR#0": PRINT
      : PRINT : VTAB 22: PRINT "EN
      TER MEMO DATE TO ";: INVERSE
      : PRINT "DELETE";: NORMAL : PRINT
      " (DD/MM/YY)": PRINT "OR,<RE
      TURN> FOR MEMU";: HTAB 27: INPUT
      ":";FL$: IF FL$ = "" THEN 22
      0
1450  PRINT D$;"DELETEAPP:";FL$: HTAB
      1: VTAB 22: PRINT "  THE FIL
      E IS DELETED": FOR P = 1 TO
      1000: NEXT : GOTO 200
1460  REM
1470  REM  QUIT
1480  GOSUB 1560: HOME : END
1490  REM
1500  REM  INPUT TECH.SUB.
1510 M$ = "":H$ = M$: INVERSE : HTAB
      Z: PRINT "      ";: HTAB Z +
      2: NORMAL : PRINT ":";: INVERSE
      : HTAB Z: FOR I = 0 TO 1: GET
      A$: PRINT A$;:H$ = H$ + A$: NEXT
      : HTAB Z + 3: FOR I = 0 TO 1
      : GET A$: PRINT A$;:M$ = M$ +
      A$: NEXT : NORMAL : HTAB 24:
      PRINT "<RETURN>=CONFIRM";: GET
      A$: IF A$ =  CHR$ (13) THEN
       RETURN
1520  GOTO 1510
1530  REM  READ APPOINTMENT
1540 N = 0: HTAB 1: VTAB 22: PRINT
      "  LOAD APPOINTMENT DATE ";D
      T$; TAB( 39): POKE 34,23: PRINT
      D$;"OPENAPP:";DT$;",S";DS;",
      D";DN: PRINT D$;"READAPP:";D
      T$: INPUT N: FOR I = 0 TO N:
       INPUT H$(I): INPUT M$(I): INPUT
      A$(I): NEXT : PRINT D$;"CLOS
      E": POKE 34,0: RETURN
1550  REM  WRITE APPOINTMENT
1560  IF N <  = 3 THEN  VTAB 22: HTAB
      1: PRINT "  NO APPOINTMENT..
      DELETE OPENED FILE": POKE 34
      ,23: PRINT D$;"DELETEAPP:";D
      T$;",S";DS;",D";DN: POKE 34,
      0: VTAB 22: HTAB 1: PRINT  TAB(
      39): RETURN
1570  HTAB 1: VTAB 22: PRINT "  S
      AVE CURRENT APPOINTMENT ";DT
      $; TAB( 39): POKE 34,23
1580  IF DO$ = "" THEN 1600
1590  PRINT D$;"DELETEAPP:";DT$
1600  PRINT D$;"OPENAPP:";DT$;",S
      ";DS;",D";DN: PRINT D$;"WRIT
      EAPP:";DT$: PRINT N: FOR I =
      0 TO N: PRINT H$(I): PRINT M
      $(I): PRINT A$(I): NEXT : PRINT
      D$;"CLOSE": POKE 34,0: RETURN
```

TAKE 1 同一公司出品

簡明易用的繪圖軟件包-PIXIT

張峯寧



PIXIT 是一個非常強力並且易用的繪圖軟件包。它充份發揮了蘋果高解像的繪圖功能，通過這軟件你可畫出一些專業化及高品質的圖畫。它的使用對象並不需要有豐富的電腦知識，反之，只需要有一點使用電腦的經驗便行了。這個軟件包適用於任何APPLE 系列或相容的機種。

圖畫編輯（PICTURE EDITOR）

首先BOOT起PIXIT，螢幕上會出現一個功能選擇表，見表(A)，其中包括PICTURE EDITOR，CREATE—A—SHAPE，SHAPE TABLE EDITOR，DEMO及QUIT （注意：市面上有一隻 PIXIT 版本是沒有DEMO的，但用法一樣而且DEMO用處不大）。

表A	
	PICTURE EDITOR
	CREATE—A—SHAPE
	SHAPE TABLE EDITOR
	DEMO
	QUIT

當閣下從表中選擇PICTURE EDITOR，螢幕畫面會出現另一個如圖一所示的選擇表 。

PICTURE EDITOR 能夠利用圖形表去繪畫高解像圖畫、在高解像畫面上顯示文字、畫線、圓圈等。它又能用來修改已有的圖畫，而圖形是可以有不同的顏色、比例及旋轉角度，並可以隨處移動的。

從圖(一)的選擇表中，我們可找到10項選擇，除了頭三項需要稍加說明外，其餘均十分淺白，無需多加解釋。事實上，PIXIT 的使用就如同上期介紹的TAKE 1一樣，大家只要望文便知其義，一學即會使用。PIXIT 的出品公司就是出版TAKE 1 的同一間公司。

LOAD SHAPE TABLE

PIXIT 的設計者預先造好了多種類的圖形供用者使用，這包括：

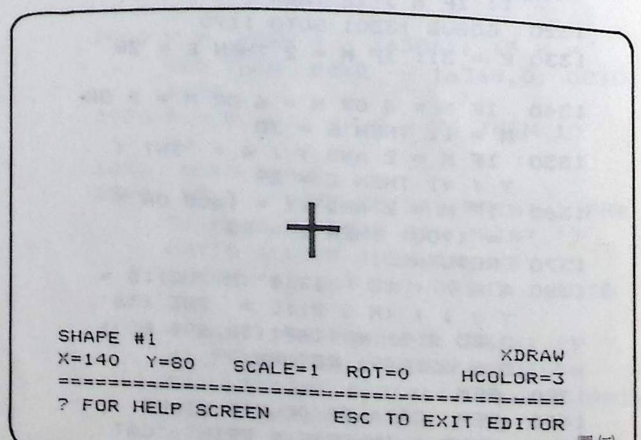
(1) TUTORIAL SHAPE
(2) ANIMALS
(3) GEOMETRICS
(4) PAINTS & PATTERNS
(5) TREE
(6) ANIMALS SHAPE

```
PIXIT PICTURE EDITOR
------------------------------------------

        FUNCTION MENU
LOAD SHAPE TABLE
LOAD CHARACTER SET
CREATE PICTURE (ERASE PICTURE IN MEMORY)
CATALOG DISK
CHANGE DISK DRIVE 1
EDIT PICTURE IN MEMORY
LOAD HI-RES PICTURE
SAVE HI-RES PICTURE
PRINTER GRAPHICS DUMP
QUIT PICTURE EDITOR

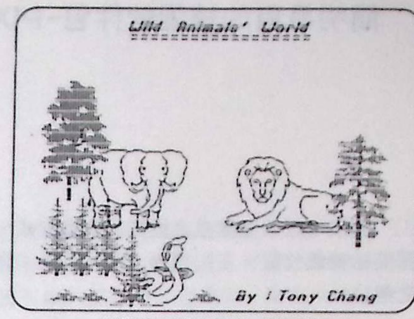
USE ARROW KEYS AND [RETURN] TO SELECT
```

圖(一)



圖(二)

簡明易用的繪圖軟件包-PIXIT



利用PIXIT軟件包輸出的一圖形例子

(7) WILD ANIMALS
(8) BUILDING
(9) GAMES SHAPE
(10) MUSIC SYMBOLS
(11) MUSIC INSTRUMENT
(12) POTPOURRI

每種類的圖形有6個至廿個之多。這些林林總總的漫畫公仔皆是非常實用。此外，作者亦內置有五種不同大小款式的字體（其中包括：BOLD 10，ITALIC 8，LARGE BL—OCK 17，BOLD 17，STANDARD 8）方便用者在寫上文字說明時使用。

CREATE PICTURE或EDIT PICTURE IN MEMORY兩種功能都能帶你進入高解像畫面。在高解像畫面下，你會見到一個閃動的十字游標及一螢幕狀態欄於底部。欄中顯示畫面之現有狀態。現逐一分析如下：（圖二）

1）SHAPE ＃：此乃記憶體中形狀表之各種不同形狀的代表號碼。如果沒有任何形狀表被輸入，則此項會被設爲置0。

2）X、Y：X值是現有游標之水平參數，Y值是現有游標的垂直參數。

3）SCALE：它指示現有圖形之相關比例，圖形可以由原來大小放大至5倍。本來圖形大小爲SCALE：1，大1倍則SCALE：2，如此類推…

4）ROT：ROT指示圖形之旋轉值，此值範圍由0至63。當ROT＝0表示圖形是正常的，沒有變動，ROT＝16時則將圖形向順時針方向轉動90度，ROT＝32是將圖形上下反轉。

5）HCOLOR：HCOLOR顯示畫面上之圖形、線或圓圈的顏色。其值範圍由0至7，代表顏色如下：

0＝BLACK 1	4＝BLACK 2
1＝GREEN	5＝ORANGE
2＝PURPLE	6＝BLUE
3＝WHITE 1	7＝WHITE 2

6）XDRAW、DRAW：XDRAW會令到畫面上之圖形以其背景之相反顏色。而DRAW 則將一圖形顯示於畫面上，不論畫面的底色是什麽顏色。

在PICTURE EDITOR中有多個指令，大家可從圖(二)中一目瞭然。其中最能帮助你的是〔？〕鍵。任何時間只需按下這個〔？〕鍵，便可將各項的控制指令列出。

有關圖(二)內各項選擇大家只要細看說明即可，無需筆者浪費筆墨。

各位在使用PICTURE EDITOR時候，大致的步驟如下

① 選定LOAD SHAPE TABLE；
② 選定CHARACTER SET；
③ 選擇CREATE PICTURE功能；
④ 開始製造過程。

製造圖形功能（CREATE-A-SHAPE）

此程式讓你可直接利用鍵盤設計高解像圖形，當然，它亦包括修改圖形的功能。首先從程式選擇表，選定CREATE—A—SHAPE 程式，會出現一個功能選擇表，包括圖四中的所列七項功能。

繪製新圖形

當各位從表中選擇CREATE NEW SHAPE功能後，第一件事要做的便是選擇繪畫比例。繪畫比例值由1～7。

筆者在此建議用家選擇較大之比例，因爲只有在這情況下，我們才可看見較精細的圖形。

接着，如果你所選擇的比例是3或以上，則程式會詢問要否格子帮助繪畫工作。在正常情况下，有格子帮助是比較好的。

```
            PICTURE EDITOR COMMAND
  FAST MOVE - X10        SLOW MOVE - X1

 UP      W               UP      I
 LEFT  A    S  RIGHT     LEFT  J    K  RIGHT
 DOWN    Z               DOWN    M

          SPACE DISPLAY SHAPE

 X XDRAW/DRAW            P PLANT SHAPE
 V SPLIT/FULL SCREEN     L DRAW LINE
 C CHANGE HCOLOR         O DRAW CIRCLE
 Y CHANGE SCALE          T ADD TEXT
 R CHANGE ROTATION       F COLOR FILL
 B BACKWARDS SHAPE       G GRAPHICS PAD

 USE ARROW KEYS AND [RETURN] TO SELECT
 SHAPE, SCALE, ROTATION, AND HCOLOR

 SHAPE #1                             XDRAW
 X=140  Y=80   SCALE=1  ROT=0   HCOLOR=3
 ========================================
 ? FOR HELP SCREEN       ESC TO EXIT EDITOR
```

圖(二)

```
          PIXIT CREAT-A-SHAPE
 ----------------------------------------

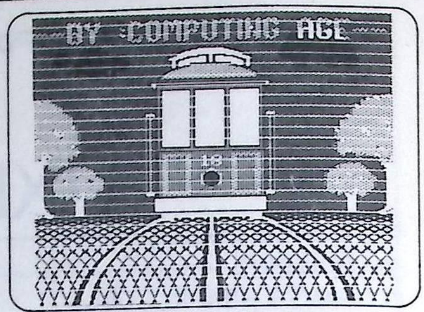
             FUNCTION MENU:

 CREATE NEW SHAPE (ERASE SHAPE IN MEMORY)
 COMPILE SHAPE TABLE
 REVIEW & EDIT SHAPE IN MEMORY
 CATALOG DISK
 CHANGE DISK DRIVE 1
 SHAPE SOURCEROR
 QUIT

 USE ARROW KEYS AND [RETURN] TO SELECT
```

圖四

簡明易用的繪圖軟件包-PIXIT



利用PIXIT軟件包繪出的一圖形例子

用家可以根據擬畫之圖形的繪畫方式、大小及用地來選擇開始繪圖位置。在經過幾次的繪畫後，你必能掌握此選擇之要訣。

完成各項後，一個繪圖畫面會出現在螢光幕上。（圖五）

請留意左下方有MOVE ＃1字，此狀態之參數是隨着你繪畫時移動次數而增加的。而一個圖形的MOVE値最多可達8000（相信閣下很難會有機會畫出MOVE値達八千之多的圖形）。

CREATE-A-SHAPE 編輯指令分析

同樣地，在CREATE—A—SHAPE中，我們可利用I、J、K、M、A、W、S、Z 等鍵來移動游標，但兩組鍵每按一下，都只是移動一步而已，而兩組鍵功能的分別在利用W、A、Z、S鍵時，只會移動，不會在畫面上畫點，利用I、J、K、M則可。

在此模式中，有多個特別功能指令鍵値得加以詳述：CTRL—B 鍵，此鍵會令游標回到檔案開始時游標的位置，並抹去畫面上的圖形。但是，這鍵並不會清除記憶體中的圖形。故，我們可使用→鍵或CTRL—E去重新繪出本來的圖形。

CTRL—E鍵：如果當游標是在檔案開始或中間時的位置，按CTRL—E會重新繪出本來圖形。此鍵只須你按一下便會自動工作，而按SPACE BAR則會中止此工作。

CTRL—X鍵：此鍵容許你在任何時候，改變繪畫比例、格子（GRID） 選擇或開始位置，但並不影响記憶中的檔案。當你做了改變工作後，繪畫畫面會重新出現，而跟着你畫的圖形都是根據你剛輸入的資料。

CTRL—F鍵：按下此指令鍵，會把當時游標位置設為完結點並抹去跟着的圖形。

當你移動到畫面邊緣企圖繼續移動繪畫時，SCREEN BOUNDARY 的訊息會出現。在這情況下，你可用CTRL—X鍵去改變繪畫比例，而繼續繪畫。

編輯圖形表

如前所述，編輯圖形表（COMPILE SHAPE TABLE）之功能是改變向量，把檔案移入APPLESOFT SHAPE TA—BLE中。

本程式會根據狀態顯示欄中圖形的開始地址及長度的—BYTES而在高解像畫面上繪出圖形。

SHAPE SOURCEROR功能之使用

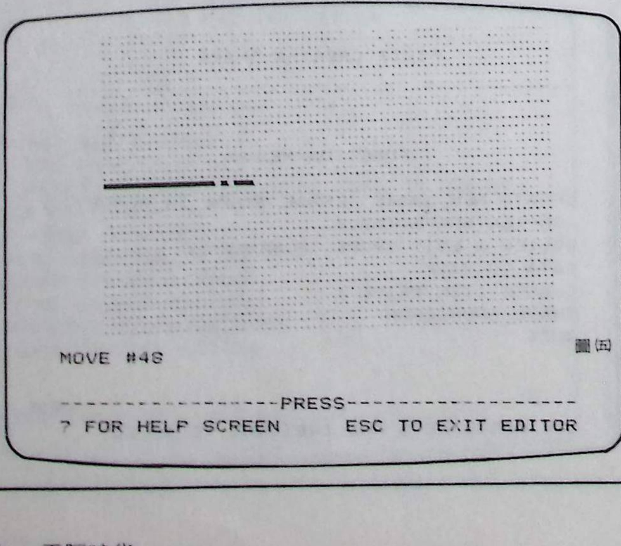
大家若果從功能選擇表中選擇進入SHAPE SOURCER—OR 功能，則程式容許你從磁碟中LOAD 入圖形表及分析修改表中之圖形。事實上，我們可利用此功能去分析PIX—IT 程式內置圖形表圖形之畫法。

首先放入儲有圖形表之磁碟，進入SHAPE SOURCER—OR 功能，選擇所需之圖形表，按RETURN 。在一個圖形表中如含有多過一個的圖形的話，SHAPE ＃1是會被顯示在高解像畫面上，你可以利用左右箭咀選擇圖形，並按下RETURN 鍵使之顯示於畫面上。如要修改圖形，可利用—CREATE—A—SHAPE 之 編輯指令進行工作。 當然，在工作完畢後，你可把圖形儲存在磁碟上。

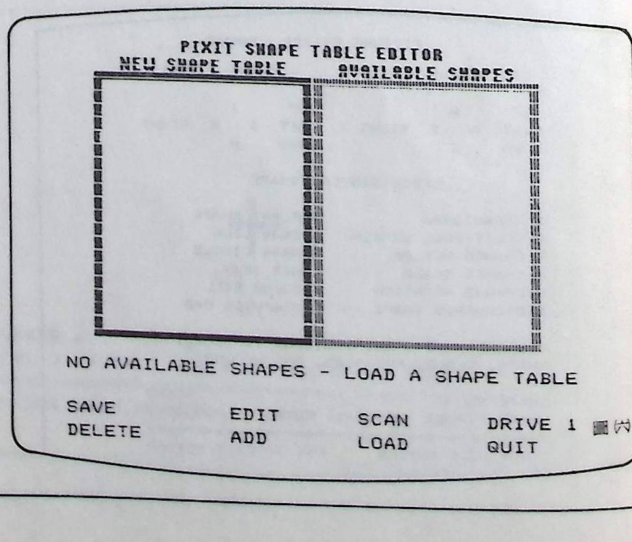
一些繪圖技巧

你應該多些練習繪畫，從而熟識多種不同的指令。在此筆者建議你先試畫一些簡單的圖形，例如，正方形、長方形。然後再進一步畫複雜的圖形。在繪畫過程中，下列兩點是你必須留意的：

1）如果在繪畫過程中，想從PLOT轉到NO—PLOT或從NO—PLOT轉到PLOT模式時，你必須肯定在改變時，要保持本來的方向。例如，你要從PLOT轉到NO—PL—OT 模式時，你要在改變時，肯定是在NO—PLOT模式下。



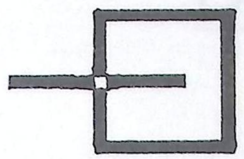
圖(五)



圖(六)

簡明易用的繪圖軟件包-PIXIT

2）如果有兩條線重叠的話，「混濁」的畫面便會產生了！——如果你在一個正方形上畫一條線，而兩者是重叠的話，則兩者相交點會是不見了！（參考下圖）



所以，要避免這問題，我們必須在重叠之前——在畫線時遇到以前所畫的線時，用NO－PLOT跨過以前所畫的線。這樣，問題就被解决了！

3）在畫完圖形後，尤其是那些有重叠的圖形，最好在COMPILING SHAPE TABLE之前，使用CTRL－B及CTRL－E去REVIEW一個圖形，如沒有任何問題，才進入COMPILING SHAPE TABLE，並將之存在磁碟上。

圖形表編輯

圖形表編輯程式是用作建造圖形表之用，而這個圖形表最多可容納128個圖形。你可以從PIXIT CREATE－A－SHAPE程式、PIXIT SHAPE LIBRARY或標準APPLE SOFT SHAPE TABLE中LOAD入圖形。

在此程式下，你可以修改在表中的圖形，也可以把幾個圖形合併爲一個。

現在載入SHAPE TABLE EDITOR 程式，畫面上會出現兩個框格，分別爲AVAILABLE SHAPES及NEW SHAPE TABLE。（圖六）

而由於此程式沒有任何附加指令，所以並沒有指令列表的存在。所有功能都顯示在螢幕底——SAVE、EDIT、DE－LETE、ADD、SCAN、LOAD、DRIVE 1、QUIT，用家可利用左右箭咀選擇功能，選定後，按RETURN執行之。

我們的主要目的是從磁碟上LOAD入圖形表於AVAIL－ABLE SHAPES 框格中，然後將圖形由此框格移到NEW SHAPE TABLE 框格中去建立自己的圖形表。在任何時候，你可以在不影响NEW SHAPE TABLE 的情況下，載入新的圖形表於AVAILABLE SHAPES 框裏。你更可以將這些圖形加在NEW SHAPE TABLE 的圖形中的任何位置。

如果你沒有載入任何圖形表的話，則在框格下會出現下列字眼——"NO AVAILABLE SHAPES－LOAD A SHAPE TABLE"，而在此情況下，除非你載入圖形表，否則你不能做任何東西。而當你載入圖形表後，在兩個框格下，都會分別出現一些新的螢幕資料。

在NEW SHAPE TABLE 框格裏會是空的，但在其下會有"SHAPE＃1 MEM：8188"等字。SHAPE＃1 及空的框格告訴你在NEW SHAPE TABLE裏的圖形位置（POSITION＃）1是空的。MEM：8188 告訴你，現在有大8K BYTES的記憶空間可供使用。而當你在格裏加上圖形時，此值便會隨圖形之增加而減少。

此外，在AVAILABLE SHAPES框格裏應該會有一圖形及在其格下有"SHAPE ＃1 SIZE：XXX"等字。SHA－PE ＃1 告訴你，格中的圖形是AVAILABLE SHAPES的第一個圖形。而"SIZE："則告訴你格中圖形以BYTES計算的大小。

SHAPE TABLE EDITOTOR 功能分析

當圖形被載入AVAILABLE SHAPES 框格後，你就可以準備開始建立NEW SHAPE TABLE 了！首先，從PIX－IT SHAPE LIBRARY或資料磁碟載入一圖形表。同樣地，按E S C鍵能中止此工作。SHAPE TABLE EDITOR 包括有多項的功能操作。（見圖六）

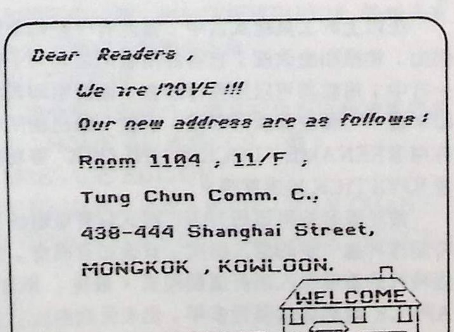
這些功能指令操作大家可輕易從程式中明白到，在此不再多解釋。

到此筆者已詳細地敍述了PIXIT的用法，相信大家都知道怎樣使用了吧！

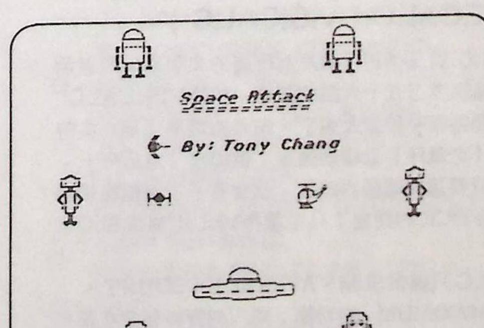
假如閣下能利用此程式再配合上期介紹的TAKE 1，則可做出很多驚人的東西來。

又筆者利用PICTURE EDITOR 造了7個DEMO，以供參考之用！

怎樣，現在還不拿出PIXIT程式，繪出自己心愛的圖形！



利用PIXIT軟件包繪出的一圖形例子



利用PIXIT軟件包繪出的一圖形例子

爲APPLE PLUS而寫的綜合軟件JANE

集會計、文字處理、數據庫處理於一身

華德

在近期推出衆多的工具程式當中，JANE 是最受歡迎的一個，JANE 所以受歡迎，原因是它包含了 VISICALC、WORDSTAR及 dBASE II 的卓越功能。而最重要一點是，JANE 是 APPLEWORK的 II PLUS 版本，是爲了照顧 II PLUS 版本的用家。

原裝JANE PROGRAM 共包括了三張不同顏色的磁碟（BLACK、GREY及YELLOW）及一本厚達二百三十六頁的說明書，雖然這本說明書非常厚，但它很容易使人明白，而程式本身只要看著螢幕便會使用。程式適宜用兩部磁碟機。

JANE 是以「視窗」的形式顯示，這令用者使用時更易明白。當 JANE 執行時，畫面上方顯示一排的圖像，移去適當的圖像便可執行該種功能，例如移去打字機的圖像，便可執行WORDSTAR 的工作。

在 JANE 的圖像中，較爲特別的是STOP 圖像，它可清除現時的功能而返回主MENU 及把現時在電腦中的資料貯存入磁碟中。

三大用途：

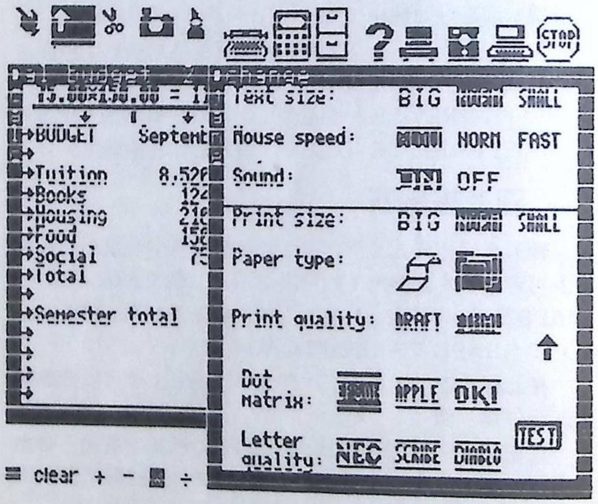
JANEWRITE（WORDSTAR）

JANEWRITE 功能提供給一些愛好用英文寫信的人士極大的方便，用者只要把JOYSTICK 移到適當的位置，然後用KEYBOARD打入你想寫的內容。至於刪改也是一樣。—JANEWRITE 可容許用者寫達二十五頁的文件或信，而—JANEWRITE 最令人讚佩的是它所顯示的文字是80COL—UMN（不需加任何CARD）。

JANECALC（VISICALC）

JANECALC 只提供19行濶及24行高的文字幕，即表示整個文字幕只提供四百五十六個的間格，較諸VISICALC 的16,000 的間格似乎相差太遠了。筆者認爲用這項功能時，最好就是用「老鼠仔」做輔助儀器，但如沒「老鼠仔」，也不要慌張，只要選擇電腦的圖像，就會有一項功能提供給用者調校JOYSTICK 的速度，只要選擇MEDIUM或SLOW便可。

JANECALC 只提供SUM、AVERAGE、COUNT、MINIMUM及MAXIMUM 的功能，輸入的資料便會在那些間格中顯示。筆者認爲整個JANE CALC 的系統對於一些私人的財政狀況會較爲實用。



JANELIST（dBASE II）

在 JANE 中，筆者最喜歡就是這項功能，它就等於一本小型的工商業或私人的電話簿，用者只要選好每項資料的字數及選擇商業或私人後，便可隨意輸入親友或僱主的姓名、電話、地址等等，用者可貯存數不勝數的電話紀錄，只要磁碟的空間夠便行，在螢幕最低的一行中，提供了不少JANE—LIST 的重要功能，好像可分類查閱資料、找尋檔案……等。

在以上的工具程式當中，當然有一些輔助的工具程式，例如：相機和塗改液，它容許用者把這句句子或資料搬到另一行中；用較剪可以把整行刪去；而箭咀即表示INSERT；印字機—可調校那款印字機、字數、輸出濶度等；磁碟—容許用者RENAME、LUCK及UNLOCK 等功能；電腦—調校JOYSTICK 的速度等。

鑑於筆者取得這個JANE 程式只有短短數天，未有足夠時間作再進一步的深入研究，日後如有機會，筆者會再借電腦時代的篇幅深入剖析這個程式。最後，筆者有一感想，APPLE 電腦雖然面世多年，仍未受到淘汰，主要是有許多許多程式設計師仍然願意爲APPLE 寫軟件，而且軟件的效能亦能跟隨時代的步伐。■

能令APPLESOFT如虎添翼

一種加强版本的BASIC語言

BEAGLE BASIC

胡永達

本人已擁有BEAGLE BASIC 多時了，但從坊間書店之中從未看見有任何書籍去介紹其用法，所以現借貴刊作小小的介紹。

BEAGLE BASIC 是一種加强版本的BASIC ，其好處是加添AP Ⅱ內缺少之指令。例如：ELSE, GOT 等，而優點是就算RESET 之後，也不會影响其工作。這個用途甚廣的軟件是BEAGLE BROS 三年前的出品。

首先介紹其主要用途：

A）BEAGLE NEWS：
這是介紹BEAGLE BROS 廠之新產品。

B）BEAGLE BASIC DEMO
這是用精簡的形式去介紹BEAGLE BASIC 之用途。

C）CREATE NEWBASIC
這是給讀者們去選擇所需要之功能。

D）NEW BASIC LOADER
這是載入預先選擇好的指令功能。

Q）QUIT
這是離開本程式。

在程式內之DEMO ，用非常生動，有趣之形式去介紹本身所有功能之用途。例如介紹ELSE在程式內之用法。

重要的部份

在本程式內，最主要的部份還是C）CREATE NEW－BASIC 。而程式內可以將APPLESOFT 的指令更改，及在每程式上加添某些功能，例如在程式內REM 的句子用反白顯示出來。以下介紹主要操作指令：

C－COMMAND EDITOR
除了DOS 指令可以更改外，連其它也可以隨讀者們喜歡而更改。例如：HOME可以改爲CLS…等。

E－ERROR MESSAGE EDITOR
它可以將所有錯誤訊息更改。例如ILLIGLE QUANTI－TY ERROR 可以改爲NUMBER TOO LARGE ERROR。

1.QUIT
2.EDITING
3.FIND
4.SORT
5.TOKEN
6.CHANGED
7.STANDARD
8.TO NORMALIZE
9.USE TO PRINTER
0.PRINT SCREEN

L－LIST FOR MATTER
可以更改LIST程式之形式。

A）INVERSE REM STATEMENT
將程式內，REM之句子，用反白形式顯示。

B）CHANGE LIST WIDE
可以更改LIST 程式之濶度。而不須要用POKE 地址33。

C）CHANGE IDENTATION
可以更改LIST程式之左右兩旁空白之濶度。

B－ADD NEW BASIC COMMANDS
可以加入新添APPLE Ⅱ內缺少之指令。

A）ADD ELSE
B）ADD SWAP
C）ADD TONE
D）ADD HSCRN
E）ADD SCRLDN
F）ADD TXT2
G）ADD GR2
H）ADD CLRKEY
I）ADD TEXT & BELL COMMANDS
J）ADD HI－RES COMMANDS
K）ADD CURSOR COMMANDS

O－OTHER FEATURES
可以更改APPLE Ⅱ內已有之指令。

A）ADDESCAPE CURSOR
B）MODIFY BELL
C）MODIFY GOTO／GOSUB SYNTAX

S－SAVE NEWBASIC
可以儲存預先CREATE之新BASIC。

Q－QUIT
離開CREATE NEWBASIC。

這個由BEAGLE BROS 所出品之BEAGLE BASIC ，如果大家能掌握其功能，比起DOS BOSS 更加受用不少。■

隨自己心意列出任何字母爲首的檔案名字

CATALOG HELPER

英傑

改動DOS 3.3的CATALOG 指令的文章有許多，電腦時代亦有刊登過多篇，花款各有不同。但似乎筆者未見過有文章或程式介紹在CATALOG 檔案時，只按自己要求單單列出用某個英文字母爲開首的檔案，例如磁碟上有數十個檔案，筆者要找的檔案是用英文字母W字頭，電腦只會將W字母爲字頭的檔案列出。這樣做法，可以節省許多時間。

爲此，筆者特別編寫了一個程式CATALOG HELPER，供大家使用。希望各位能接受。

在未介紹本程式的使用方法之前，有幾點是大家値得注意的：

(1) 此程式並沒有使用到任何DOS棄置的地方。
(2) 它不會改變任何副程式的起始位址，免了其他程式於呼叫副程式時發生錯誤。
(3) 它沒有改變CATALOG 指令的名稱。
(4) 它沒有改變第三頁的位址值（FID程式當使用CATALOG指令時是會使用第三頁的）。

當你們使用BRUN CATALOG HELPER之後，只使用CATALOG 指令，螢光幕仍會顯示所有的檔案名稱，若選用CATALOG HELPER 的特別功能，只需鍵入

CATALOG〔FILETYPE〕〔FILENAME〕

便成了。〔FILETYPE〕是檔案類型，如I，B，A或T 型檔案，若以空位鍵（SPACE BAR）代替檔案類型字符，則自動設定爲所有檔案型。〔FILENAME〕 則是檔案名稱的開首字母，若不輸入〔FILENAME〕，則自動設定爲所有檔案名稱。如此，你們便可只容許列出某類型檔案，或以特別字母爲起首的檔案。請參看例一。於列印當中，若按下任何鍵都會停止列印，直至按下另一鍵爲止，才繼續列印過程。

相信說到現時爲止，大家也會感到它比CP／M的DIR指令更爲方便吧。不過，話說回頭，CATALOG HELPER也改變了一些命令形式，若我們使用CATALOG D1，欲將第一號磁碟機的磁碟目錄列印出來時，便需要於指令中間加上一個逗點，如用CATALOG，D1或CATALOG B，D1。而“DISK VOLUME”字樣和磁冊號數也不會出現。

在鍵入CATALOG HELPER 程式入電腦時，可以直接用LISA ASSEMBLER語言鍵入程式，又或由位址＄6000開始鍵入機械碼；當鍵入完畢後，你們可以用BSAVE CATALOG HELPER，A＄6000，L＄14C 將程式SAVE入碟。當使用BRUN CATALOG HELPER後，程式會改動CATALOG 的常規程式，此時你們可以使用INIT指令將CATALOG HELPER改成爲DOS 的永久部份。

爲了加深各位對CATALOG HELPER 的了解，我們不妨看看它的工作情形。每次我們使用DOS指令時，DOS都會從位址＄A909處的有效命令鍵語表（COMMAND PARAMETER TABLE）找出是否輸入的爲DOS 指令，並查知該指令是否需要一個檔案名稱，因此我們只要改動位址＄A929 由值＃＄40改爲＃＄60，DOS便會設定輸入的CATALOG 指令需要一個檔案名稱，並有效地將檔案名稱儲存入主檔名緩冲區（PRIMARY NAME BUFFER）於位址＄AA75至＄AA92處。

爲確實使用CATALOG 指令時，必隨著一個檔案名稱（若沒有檔案名稱我們會收到錯誤訊息），我們將CATALOG指令改爲CATALO，這個新的指令名稱將儲存於位址＄A884處的DOS命令名稱主文表（COMMAND NAME TABLE）。爲了確保程式全無錯漏，主文表將以標準的命令名稱被重寫一次，除了CATALOG 指令。亦即是說，除了將CATALOG 指令名稱改爲CATALO外，以前若你們用DOIT代替RUN指令，又或是以AP代替APPEND 指令都不再容許，請留意。

你們或許會奇怪，若此時我們使用CATALOG 指令，豈不是在尋找用G型檔的檔案？非也，雖然此時DOS會將一個以G爲字母的檔案名稱載入主檔名緩冲區，但這個字程式會忽略不管的，舉另一個例子，我們輸入指令CATALOG，GAT會被DOS載入主檔名緩冲區，此時程式會避開第一個ASCII 碼，將第二個和第三個ASCII 碼與檔案型號和檔案名稱比較，若相乎，便把檔案名稱列印出來，如此的處理方法，便一切沒有問題了。

在檔案的處理方面，CATALOG HELPER 可以自由進出檔案處理程式（FILE MANAGER），雖然檔案處理程式每次找尋一個檔案名稱之後，都會設定有一個檔案被開啓，亦即是說它會佔用一個檔案緩冲區被使用，如果每次開啓一個檔案後都加以保留，則會因有限的緩冲區而宣告“NO BUFFER AVAILABE”ERROR。爲了免去這種錯誤，程式以JSR ＄A764將開啓了的檔案緩冲區關閉，在檔案緩冲區的開頭處放入＃＄00。

最後，程式以JSR ＄A095清除主檔名緩冲區，因而使得任何使用第三頁的程式會於第一次列印受到破壞。另一點要各位注意的是，在FID程式使用CATALOG 命令時只會將INTEGER BASIC 型檔案以字母“D”爲開首的檔案名稱列印出來，這是此程式的唯一缺陷，其餘一切是如常操作的，各位大可放心使用。

若各位想將CATALOG HELPER 改動的話，有幾點是要提醒各位的：

(1) CATALOG HELPER 是佔有舊有的CATALOG 常規程式位置，故此於加入任何新程式時，只可在程式中做功夫，不可直接加在程式後面。
(2) 而CROUT副程式的起始位址仍在＄AE2F 處，你們可以如常呼叫使用。
(3) 在呼叫位址＄AE42處的PRINLEN 程式之前，在位址＄44，＄45處是放有檔案長度的低位元和高位元組，以使顯示檔案長度時可大於值255。■

CATALOG HELPER

```
!A
**END OF PASS 1

**END OF PASS 2

0800              ;
0800              ;    CATALOG.HELPER
0800              ;  By Computing Age
0800              ;  Copyright (c) 1985
0800              ;     LISA Assembler
6000                       ORG $6000
6000                       OBJ $800
6000              ;
6000              ;    *** ZERO PAGE LOCATIONS
6000              ;
0040              BUFFPTR  EPZ $40
0044              LENCALO  EPZ $44
6000              ;
6000              ;    *** DOS LOCATIONS
6000              ;
A095              CLRBUFFS EQU $A095            ; Clear buffers.
A764              FINDBUFF EQU $A764            ; Locate open buffer.
A884              KEYTABLE EQU $A884            ; DOS cmd name table.
A929              PARAM    EQU $A929            ; Catalog paramater H.
AA66              VOLSAVE  EQU $AA66            ; Last volume number.
AA77              SOUGHT   EQU $AA77            ; File/name sought.
ABDC              INITFM   EQU $ABDC            ; Init fmgr workarea.
AD98              CATLOG   EQU $AD98            ; Start of Catalog.
AE42              PRINLEN  EQU $AE42            ; Print 2hex as 3dec.
AFF7              INVTOC   EQU $AFF7            ; Read/write the VTOC.
B011              GETSEC   EQU $B011            ; Read a dir sector.
B230              ADVINDX  EQU $B230            ; Advance to next file.
B37F              FMGRXT   EQU $B37F            ; Exit thru filemgr.
B39C              DIRDEX   EQU $B39C            ; Directory index.
B3A7              TYPNAM   EQU $B3A7            ; File type letters.
B4C6              DIRTRK   EQU $B4C6            ; Track of T/S list.
B4C8              FITYPE   EQU $B4C8            ; File type & lock bit.
B4E7              FISIZE   EQU $B4E7            ; File size in sectors.
B5F9              VOLNUM   EQU $B5F9            ; Volume number current.
6000              ;
6000              ;    *** MONITOR LOCATIONS
6000              ;
C000              KEYBRD   EQU $C000
C010              KEYSTRB  EQU $C010
FDED              COUT     EQU $FDED
FDOC              KEYIN    EQU $FDOC
6000              ;
6000              ;    *** CONSTANTS
6000              ;
0000              ZERO     EQU $00
008D              CR       EQU $8D
00A0              SPACE    EQU $A0
00FF              ZROCMP   EQU $FF              ; Zero's Complement.
6000              ;
6000              ; RELOCATION ROUTINE:
6000              ;
6000 A2 A9                 LDX #1+END-START
6002 BD A4 60     RLOOP    LDA START,X
6005 9D 98 AD              STA CATLOG,X
6008 CA                    DEX
6009 E0 FF                 CPX #ZROCMP
600B D0 F5                 BNE RLOOP
600D              ;
600D              ; CHANGE  CATALOG TO CATALO
600D              ;
600D A2 00                 LDX #ZERO
600F BD 20 60     CLOOP    LDA KEYWORDS,X
6012 F0 06                 BEQ PARAMO
6014 9D 84 A8              STA KEYTABLE,X
6017 E8                    INX
6018 D0 F5                 BNE CLOOP
601A              ;
601A              ; CHANGE CATALOG PARAMATER TO ACCEPT FILENAME
601A              ;
601A A9 60        PARAMO   LDA #$60
601C 8D 29 A9              STA PARAM
601F 60                    RTS
6020              ;
6020 49 4E 49     KEYWORDS DCI 'INIT'           ; Hi bit set.
6023 D4
6024 4C 4F 41              DCI 'LOAD'           ; on last letter.
6027 C4
6028 53 41 56              DCI 'SAVE'
602B C5
602C 52 55 CE              DCI 'RUN'
602F 43 48 41              DCI 'CHAIN'
6032 49 CE
6034 44 45 4C              DCI 'DELETE'
6037 45 54 C5
603A 4C 4F 43              DCI 'LOCK'
603D CB
603E 55 4E 4C              DCI 'UNLOCK'
6041 4F 43 CB
6044 43 4C 4F              DCI 'CLOSE'
6047 53 C5
6049 52 45 41              DCI 'READ'
604C C4
604D 45 58 45              DCI 'EXEC'
6050 C3
6051 57 52 49              DCI 'WRITE'
6054 54 C5
6056 50 4F 53              DCI 'POSITION'
6059 49 54 49
605C 4F CE
605E 4F 50 45              DCI 'OPEN'
6061 CE
6062 41 50 50              DCI 'APPEND'
6065 45 4E C4
6068 52 45 4E              DCI 'RENAME'
606B 41 4D C5
606E 43 41 54              DCI 'CATALO'
6071 41 4C CF
6074 4D 4F CE              DCI 'MON'
6077 4E 4F 4D              DCI 'NOMON'
607A 4F CE
607C 50 52 A3              DCI 'PR#'
607F 49 4E A3              DCI 'IN#'
6082 4D 41 58              DCI 'MAXFILES'
6085 46 49 4C
6088 45 D3
608A 46 D0                 DCI 'FP'
608C 49 4E D4              DCI 'INT'
608F 42 53 41              DCI 'BSAVE'
6092 56 C5
6094 42 4C 4F              DCI 'BLOAD'
6097 41 C4
6099 42 52 55              DCI 'BRUN'
609C CE
609D 56 45 52              DCI 'VERIFY'
60A0 49 46 D9
60A3 00                    HEX 00
60A4              ;
60A4              START:
60A4 20 DC AB              JSR INITFM           ; Init fmgr workspc.
60A7 A9 FF                 LDA #ZROCMP
60A9 8D F9 B5              STA VOLNUM           ; Set vol#=0 comp.
60AC 20 F7 AF              JSR INVTOC           ; Read the VTOC.
60AF              ;
60AF 20 2F AE              JSR CROUT            ; or $BA69 w/Free Space.
60B2              ;
60B2 18                    CLC                  ; For 1st sector.
60B3 20 11 B0     NEXTSEC  JSR GETSEC           ; Get next sector.
60B6 B0 75                 BCS ALLDONE          ; Done? Yes, exit.
60B8 A2 00                 LDX #ZERO            ; For start of sector.
60BA 8E 9C B3     CATLOOP  STX DIRDEX           ; Save current loc.
60BD BD C6 B4              LDA DIRTRK,X         ; Get track number.
60C0 F0 6B                 BEQ ALLDONE          ; Zero? Yes, exit.
60C2 30 62                 BMI TESTDONE         ; Deleted? Yes, skip.
60C4 BD C8 B4              LDA FITYPE,X         ; Get the file type.
60C7 08                    PHP                  ; Save N-flag status.
60C8 29 7F                 AND #$7F             ; and strip lock bit.
60CA A0 07                 LDY #$07
60CC 0A                    ASL
60CD 0A           TYPELOOP ASL                  ; Shift on-bit to get
60CE B0 03                 BCS GOTTYPE          ; offset to type with
60D0 88                    DEY                  ; offset in y-reg.
60D1 D0 FA                 BNE TYPELOOP
60D3 B9 A7 B3     GOTTYPE  LDA TYPNAM,Y         ; Get Character and
60D6 A8                    TAY                  ; save it.
60D7 CD 76 AA              CMP SOUGHT-1         ; File type sought?
60DA F0 07                 BEQ CHKLETR
60DC A9 A0                 LDA #SPACE           ; No, a space?
60DE CD 76 AA              CMP SOUGHT-1
60E1 D0 43                 BNE TESTDONE         ; No, skip.
60E3 AD 77 AA     CHKLETR  LDA SOUGHT           ; Yes, name sought?
60E6 DD C9 B4              CMP DIRTRK+3,X
60E9 F0 04                 BEQ CHKLOCK          ; No, a space?
60EB C9 A0                 CMP #SPACE
60ED D0 37                 BNE TESTDONE         ; No, skip.
60EF A9 A0        CHKLOCK  LDA #SPACE           ; Yes, locked?
60F1 28                    PLP
60F2 10 02                 BPL NOLOCK           ; No, print space.
60F4 A9 AA                 LDA #"*"             ; Yes, print "*".
60F6 20 ED FD     NOLOCK   JSR COUT
60F9 98                    TYA                  ; Restore type letter.
60FA 20 ED FD              JSR COUT             ; and print it.
60FD A9 A0                 LDA #SPACE
60FF 20 ED FD              JSR COUT             ; Print a space.
6102 BD E7 B4              LDA FISIZE,X
6105 85 44                 STA LENCALO          ; Print length.
6107 BD E8 B4              LDA FISIZE+1,X
610A 85 45                 STA LENCALO+1
610C 20 42 AE              JSR PRINLEN
610F A9 A0                 LDA #SPACE           ; Print a space.
6111 20 ED FD              JSR COUT
6114 E8                    INX
6115 E8                    INX
6116 E8                    INX
6117 A0 1D                 LDY #$1D             ; Y-reg for 30 letters.
6119 BD C6 B4     NAMELOOP LDA DIRTRK,X
611C 20 ED FD              JSR COUT             ; Print file name.
611F E8                    INX
6120 88                    DEY
6121 10 F6                 BPL NAMELOOP
6123 20 2F AE              JSR CROUT
6126 20 30 B2     TESTDONE JSR ADVINDX          ; Get next file.
6129 90 8F                 BCC CATLOOP          ; More? Yes, again.
612B B0 86                 BCS NEXTSEC          ; No, next sector.
612D              ;
612D 20 64 A7     ALLDONE  JSR FINDBUFF         ; Find opened file.
6130 A2 00                 LDX #ZERO
6132 8A                    TXA
6133 81 40                 STA (BUFFPTR,X)      ; and close it.
6135 20 95 A0              JSR CLRBUFFS         ; Clear buffers.
6138 4C 7F B3              JMP FMGRXT           ; Exit thru filemgr.
613B              ;
AE2F              CROUT    EQU $AE2F
613B A9 8D                 LDA #CR              ; Carriage return.
613D 20 ED FD              JSR COUT
6140 AD 00 C0              LDA KEYBRD           ; Key pressed?
6143 10 06                 BPL CROUTO
6145 8D 10 C0              STA KEYSTRB          ; Yes, clear strobe,
6148 20 0C FD              JSR KEYIN            ; and pause.
614B 60           CROUTO   RTS                  ; No, return.
614C              ;
614C              END      END

***** END OF ASSEMBLY
```

編印有結構性的程式列表，提高偵錯時的可讀性

LIST大師

一個程式設計師編寫APPLESOFT 程式時，如果利用結構性編程技巧，則日後偵錯或測試該程式時會較為方便容易。

在偵錯或改寫程式時，假如能將程式作有系統性列出在螢幕或印字機上，加強其可讀性效果，例如將每個新指令重新開一行，指出FOR—NEXT迴路及IF—THEN 敍述句、將REM 敍述句用一行星號間開，則對於偵錯該程式會有莫大的幫助。

各位可參考圖(一)及圖(二)兩個例子列表。圖(一)是一個用普通LIST 指令列出的程式列表，圖(二)則是經過一個即將會為大家介紹工具程式〔LIST大師〕處理列出的列表。只要大家作一比較，便會知道那種形式的列表可讀性高。

〔LIST 大師〕是一個機械語言寫成的程式，功能是在LIST一個APPLESOFT 程式時能作一有結構系統性的列出，方便我們檢讀。

使用這程式十分簡單，只要BRUN LISTING MASTER〔本程式檔案名〕即可。當LOAD入一個APPLESOFT 程式並要將它LIST 出來時，只要按入&〔RET〕即可，無須打入LIST指令。假如只LIST 某行號或某段行號亦可，例如& 20，則只會將行句20列出；& 20，100 會將行句20至100之間的各行列出。假如要輸出往印字機，則只消打入PR #n (SLOT NO.) & 即可。本程式的優異處是保留了CT—RL S (暫停) 及CTRL C (中止列表) 的兩個指令。

鑑於APPLESOFT 所放置變數的位置通常會被本程式所佔用著，假如想執行，該APPLESOFT 程式，則宜先打入HIMEM：37632〔RET〕，此舉便不會因APPLESOFT程式內的變數執行後會覆蓋了本程式。

```
10   REM   ***********************
20   REM   *       MATCHING       *
30   REM   *    BY   COMPUHOUSE   *
40   REM   * COPYRIGHT  (C)  1985 *
50   REM   * BY COMPUTING AGE     *
60   REM   *                      *
70   REM   ***********************
80   REM
90   LOMEM: 24576: GOTO 760
100   REM
110 II = II - 1:TT =  INT (II / 1
    5): IF II / 15 <   > TT THEN
     RETURN
120  &  POP 75,2:J = TT > 9: IF T
    T = 5 THEN FF = 112
130  POKE 1892,TT - 10 * J + FF: IF
    TT < 20 THEN  POKE 1891,FF +
    J
140  IF TT THEN  RETURN
150  OF = 1: POP : RETURN
```

圖(一)

```
****************************************
10     REM  **********************
20     REM  *      MATCHING      *
30     REM  *   BY  COMPUHOUSE   *
40     REM  * COPYRIGHT (C) 1985 *
50     REM  * BY COMPUTING AGE   *
60     REM  *                    *
70     REM  **********************
80     REM
----------------------------------------
90     LOMEM: 24576:
       GOTO 760
****************************************
100    REM
----------------------------------------
110    II = II - 1:
       TT =  INT (II / 15):
       IF II / 15 <  > TT
          THEN  RETURN

120    &  POP 75,2:
       J = TT > 9:
       IF TT = 5
          THEN FF = 112
130    POKE 1892,TT - 10 * J + FF:
       IF TT < 20
          THEN  POKE 1891,FF + J
140    IF TT
          THEN  RETURN

150    OF = 1:
       POP :
       RETURN
```

圖(二)

或者你們已擁有類似功能的工具程式，不過筆者相信，這些程式大部份是用APPLESOFT 編寫，在執行時並不能直截了當。

輸入本程式相當簡單：大家只要CALL—151〔RET〕進入MONITOR，然後由位置9300開始。打完後用下列指令SAVE入碟：

BSAVE LISTING MASTER，A＄9000，L＄243。

本程式是用DOS TOOL KIT內的ASSEMBLER寫成。這裏只列出經組合後的OBJ CODE。至於原程式因會佔用頗長篇幅，故電腦時代編輯提議將SOUCRE CODE 放入今期出版的磁碟上，各位有興趣深入研究本程式的讀者可從磁碟上取得這個原始碼程式。

最後要一提大家，在編寫程序時，寫REM 句子，擬先以POKE 33，32 改變文字幕顯示濶度。如此，日後用本程式LIST出REM 句時，可保持同一直行，保持視覺上的美觀感。■

LIST 大師

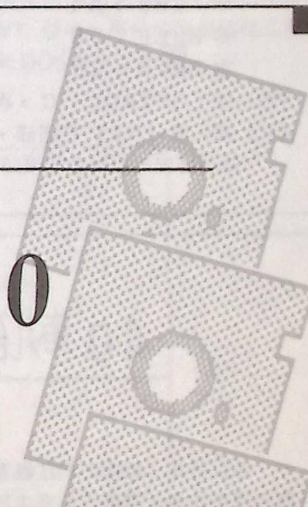
```
******************
* LISTING MASTER *
*  A$9300,L$243  *
******************
9300- A9 20 8D F5 03 A9 10 8D
9308- F6 03 A9 93 8D F7 03 60
9310- 90 0A F0 08 C9 C9 F0 04
9318- C9 2C D0 13 20 0C DA 20
9320- 1A D6 20 B7 00 F0 10 C9
9328- C9 F0 04 C9 2C D0 29 20
9330- B1 00 20 0C DA D0 21 A9
9338- 05 8D 3D 95 A9 00 8D 3F
9340- 95 8D 3C 95 8D 3E 95 68
9348- 68 A5 50 05 51 D0 0A A9
9350- FF 85 50 85 51 4C 59 93
9358- 60 A0 01 B1 9B F0 66 20
9360- 58 D8 20 FB DA C8 B1 9B
9368- AA C8 B1 9B C5 51 D0 04
9370- E4 50 F0 02 B0 4F 84 85
9378- 20 6F 94 20 24 ED AD 3D
9380- 95 85 24 A4 85 C8 B1 9B
9388- 30 02 E6 24 A4 85 4C A9
9390- 93 A4 85 29 7F 20 5C DB
9398- A5 24 CD 3B 95 90 0A 20
93A0- FB DA AD 3D 95 85 24 E6
93A8- 24 C8 B1 9B 20 30 94 AD
93B0- 42 95 D0 F5 B1 9B D0 1F
93B8- A8 B1 9B AA C8 B1 9B 86
93C0- 9B 85 9C D0 94 A9 0D 20
93C8- 5C DB 4C D2 D7 C8 D0 02
93D0- E6 9E B1 9D 60 D0 BA 10
93D8- BC 20 CF 94 38 E9 7F AA
93E0- 84 85 A0 D0 84 9D A0 CF
93E8- 84 9E A0 FF CA F0 07 20
93F0- CD 93 10 FB 30 F6 A9 20
93F8- 20 5C DB A5 24 18 69 06
9400- CD 3B 95 90 0B 20 FB DA
9408- AD 3D 95 18 69 02 85 24
9410- 20 CD 93 30 05 20 5C DB
9418- D0 F6 20 5C DB AD 3F 95
9420- F0 0A A9 00 8D 3F 95 A9
9428- 0D 20 5C DB A9 20 D0 A5
9430- 48 8A 48 98 48 A9 00 8D
9438- 42 95 B1 9B C9 3A D0 29
9440- 20 5C DB 20 FB DA E6 85
9448- A9 01 8D 42 95 C8 B1 9B
9450- 30 09 AD 3D 95 85 24 E6
9458- 24 D0 0E C9 B2 D0 05 A9
9460- 01 8D 3C 95 AD 3D 95 85
9468- 24 68 A8 68 AA 68 60 48
9470- 8A 48 98 48 A4 85 C8 B1
9478- 9B C9 B2 D0 11 AD 3C 95
9480- D0 30 A9 2A 20 9F 94 A9
9488- 01 8D 3C 95 D0 24 AD 3C
9490- 95 F0 1F A9 2D 20 9F 94
9498- A9 00 8D 3C 95 F0 13 AE
94A0- 3B 95 8D 40 95 AD 40 95
94A8- 20 5C DB CA D0 F7 20 FB
94B0- DA 60 AE 3E 95 F0 B2 AD
94B8- 3D 95 38 E9 03 CA D0 FA
94C0- C9 05 B0 02 A9 05 8D 3D
94C8- 95 8A 8D 3E 95 F0 9A 48
94D0- 8D 41 95 AA 48 98 48 AD
94D8- 41 95 C9 81 D0 0B AD 3D
94E0- 95 18 69 03 8D 3D 95 D0
94E8- 80 C9 82 D0 2A A2 01 C8
94F0- B1 9B C9 3A F0 0B C9 00
94F8- F0 07 C9 2C D0 F1 E8 D0
9500- EE AD 3D 95 38 E9 03 CA
9508- D0 FA C9 05 B0 02 A9 05
9510- 8D 3D 95 85 24 D0 D0 C9
9518- C4 D0 15 A9 0D 20 5C DB
9520- AD 3D 95 18 69 03 8D 3D
9528- 95 85 24 EE 3E 95 D0 B7
9530- C9 B1 D0 B3 A9 01 8D 3F
9538- 95 D0 AC 27 00 00 00 00
9540- 00 00 00 ■
```


COMPUHOUSE 最新出品軟件

CP/M-FORMAT 40
套裝軟件

大喜訊

CP/M用戶可將磁碟容量提高15%



每套附精裝MENU 只售50元

特約經銷處：

電腦時代讀者服務部

百老滙銅鑼灣分店（三越百貨公司對面）

百老滙旺角分店（凱聲戲院隔鄰）

COMPUHOUSE出品 電腦時代出版社總代理

41軌磁碟——你現有的磁碟所能取得的最大記憶空間

PATRICK CHAN

如果大家有看過第十三期的電腦時代的話，相信大家都參閱過麥少文君的保護程式技巧之一搬移 VTOC 法一文，其中麥君曾提及製造40軌及41軌磁碟的方法，不過筆者曾按照麥君的方法去嘗試，結果發覺磁碟的確能成爲41軌，但當作者想將程式檔案放入磁碟時，發覺磁碟上的空間和一隻普通35軌磁碟根本完全一樣，原因其實很簡單，在麥君一文中所提及的方法，只是把 VTOC 搬至40或41軌上，目的只是保護磁碟而不是要利用41軌磁碟上的空間來存放檔案。

要利用41軌磁碟上的空間，INIT 磁碟的過程便要改變一下，方法如下：

(一) BOOT 起普通的 HELLO 程式（最好是 DOS 3.3或 PRONTO－DOS，尤其後者，因爲它所佔的空間比較少）。

(二) 按入 CALL－151（進入監督系統），然後按 RETURN。

(三) 接着按入以下的命令：

*AEB5：A4

*B3EF：29

*BEFE：29

(四) 按 CONTROL－C，再按 RETURN。

(五) 插入要 INIT 的磁碟，然後按入INIT HELLO。

這樣你便可以完全利用這隻41軌磁碟上的所有空間了。

不過，你要知道磁碟機本身是不是能夠寫讀40軌磁碟，一般可以寫讀40軌磁碟的磁碟機都是可以寫讀41軌磁碟的。使用41軌磁碟有幾點是要注意的，要將檔案放入41軌磁碟最好是用 FID，因爲很多可以處理檔案程式的拷貝程式都假定磁碟是35軌的，所以，TOTAL SECTOR 的顯示都是 560，但 FID就沒有這缺點。一般普通檔案程式都可以在41軌磁碟上運行，但 COPYA 却有問題，要將 COPYA 放在41軌磁碟上運行，首先要將 COPYA 及 COPY.OBJ 0 抄到41軌磁碟上，然後按BLOAD COPY.OBJ 0及 RETURN。跟着按 CALL－151及 RETURN。然後按入*0302：29

*035F：29

*BSAVE COPY.OBJ 0，A$2C0，L$10B

再按 CONTROL－C及RETURN。這樣便可以了。

要拷貝41軌磁碟，比較方便的方法是抄檔案的方式或用 DISK MUNCHER 2·0。如用後者，只須將磁碟由００軌抄至２２軌改爲００軌抄至２８軌（即第41軌）。

最後，提醒大家一點，只是我們不能 BOOT 起41軌磁碟後立即 INIT 另一磁碟，這樣你只會得到一隻 35 軌的磁碟，我們必須根據上述的方法再做一次才可再 INIT 到另一隻41軌磁碟，希望各位留意。■

40軌磁碟SECTOR使用情況顯示

各位朋友，相信你們在觀看磁碟目錄時很想知道此磁碟尚餘多少FREE SECTOR 以供使用。可能你的HELLO程式在BOOT 機時已顯視在螢幕上，但這種方法會多佔用２個SECTORS，同時阻慢 BOOT 機的時間。筆者現在提供一程式，各位只須按下列步驟執行：

(1)鍵入程式

(2)RUN 程式

(3)鍵入NEW

(4)鍵入

```
10 HOME
20 ?"AUTO FREE SECTORS"
30 ? CHR$(4);"CATALOG"
```

(5)放入新磁碟，鍵入INIT HELLO

現在你可以關機，試開機BOOT 這隻磁碟，你會看到FREE SECTORS 顯視在螢幕的右上。如果你想其他有FILE 的磁碟擁有上述功能，只要用—LOCKSMITH 5.0或COPY Ⅱ＋3.0 將第 0 和第一條磁軌COPY就可以了。

此程式適用於40 TRACKS 磁碟，如果是35 TRACKS 只須把DATA……160，160，185…改爲DATA………160，140，185…就可以了。

```
10 X =  PEEK (978):Y = 256 * X +
    6579
20  FOR I = Y TO Y + 69: READ Z: POKE
    I,Z: NEXT
30  DATA   162,7,32,74,249,134,65
    ,160,160,185,242,179,240,10,
    10,144,251,232,208,250,230,6
    5,208,246,136,208,238,165,65
    ,172,6,224,240,5
40  DATA  32,27,229,48,3,32,36,23
    7,160,243,185,249,181,32,237
    ,253,200,208,247,76,47,174,0
    ,160,198,210,197,197,160,211
    ,197,195,212,207,210,211
50  POKE Y + 11,X + 22: POKE Y +
    46,X + 24: POKE Y + 55,X + 1
    7: POKE Y - 2287,179: POKE Y
     - 2286,X + 25
60  END
```

■

D3.3 OS 專欄

甘貴祥

兩個新DOS 3.3指令

MREAD & MWRT（下）

在第15期筆者曾經介紹了MREAD指令，今期會介紹另一個指令MWRT，它與MREAD指令相若，但MWRT會將一個＄04代碼經FILE MANAGER 送出再發出BSAVE 命令貯存資料到磁碟上。

MWRT程式會較MREAD長，主要是因爲它寫出檔案時，FILE MANAGER 的參數表內之長度須比實際寫出的長度少1，而這個長度資料是存放在＄AEA0至＄AEAD之上。

在使用MWRT命令時亦需要加進(A)檔案開始位址，(L)檔案長度。不過SLOT及DRIVE參數可以在有需要時加入，除此以外MWRT命令需加入一個新的參數欄(B)，它可以使資料貯存到磁碟上早已存有的檔案中任意位置，例如MWRT TEST，A＄2000，L＄5，B＄3會將記憶體內由＄2000開始至＄2004的資料貯存到磁碟上已經存有的TEST檔案，由TEST 檔案上的第3個位元組開始貯存，第一及第二個位元組內容則不會改變。

以下列出的兩程式與第15期所列出的兩個已被增刪修訂過，不過在執行安插兩個新指令爲DOS之時只需按入RUN INSTALL MREAD／MWRT 即可。另一個原始碼程式列表是經修改過，包含有兩個新DOS指令了。■

```
:ASM
*
*
*      FAST DOS ENHANCEMENT
*         BY ALBERT NG
*     COMPUTING AGE 1985 (C)
*
*
            ORG  $AE8E
*
*
*    CODE FOR MREAD COMMANDS
*
*    INIT & CHAIN COMMANDS
*      WILL BE REPLACED
*
*
*
SYNERR      EQU  $A000
OPEN        EQU  $A2A8
BLOAD       EQU  $A385
BSAVE       EQU  $A354
POSN        EQU  $A53B
```

```
KEYFLGS  EQU  $AA65
LEN      EQU  $AA6C
FMSUB    EQU  $B5BC
FMOP     EQU  $B5BB
FMLEN    EQU  $B5C1
*
*
* MREAD - this command will load
* any file into memory.
* A & L parameter are required
* S, D, V, B parms are optional
*
AE8E: A9 03                  LDA  #$03
AE90: 8D EC AE               STA  OPCODE
AE93: 20 B8 AE               JSR  COMCODE
AE96: B0 51                  BCS  ERR
AE98: 4C 85 A3               JMP  BLOAD
* MWRT - this command will allow
* writting to any type of
* file from memory. a & l parms
* are required.
*
*
AE9B: A9 04                  LDA  #$04       ;FM WRITE CODE
AE9D: 8D EC AE               STA  OPCODE     ;TEMP STORAGE
AEA0: 38                     SEC
AEA1: AD 6C AA               LDA  LEN
AEA4: E9 01                  SBC  #$01
AEA6: C9 FF                  CMP  #$FF
AEA8: D0 03                  BNE  LAEAD
AEAA: CE 6D AA               DEC  LEN+1
AEAD: 8D 6C AA      LAEAD    STA  LEN
AEB0: 20 B8 AE               JSR  COMCODE
AEB3: B0 34                  BCS  ERR
AEB5: 4C 54 A3               JMP  BSAVE
*
*
*
*
*
* commonly used by mread command
* and check for A & L parameter
*
* open the file & positions if
* necessary.
*
*
AEB8: A9 09         COMCODE  LDA  #$09
AEBA: 2D 65 AA               AND  KEYFLGS
AEBD: C9 C9                  CMP  #$09
AEBF: F0 02                  BEQ  L1035
AEC1: 38                     SEC
AEC2: 60                     RTS
                    L1035
AEC3: 20 A8 A2               JSR  OPEN
AEC6: AD 65 AA               LDA  KEYFLGS
AEC9: 29 02                  AND  #$02
AECB: F0 03                  BEQ  RWFILE
AECD: 20 3B A5               JSR  POSN
                    RWFILE
AED0: AD EC AE               LDA  OPCODE
AED3: 8D BB B5               STA  FMOP
AED6: A9 02                  LDA  #$02
AED8: 8D BC B5               STA  FMSUB
AEDB: AD 6C AA               LDA  LEN
AEDE: 8D C1 B5               STA  FMLEN
AEE1: AD 6D AA               LDA  LEN+1
AEE4: 8D C2 B5               STA  FMLEN+1
AEE7: 18                     CLC
AEE8: 60                     RTS
*
*
* print syntax error if address
* and length parms were absent.
*
*
                    ERR
AEE9: 4C 00 A0               JMP  SYNERR
*
*
* temporary storage for
* file manager opcode.
*
*
                    OPCODE
AEEC: 00                     BRK                  ; temp storage
                    PATCH
AEED: E0 00                  CPX  #$00
AEEF: F0 04                  BEQ  SAVEIND
AEF1: A2 02                  LDX  #$02
AEF3: D0 02                  BNE  RTFM
                    SAVEIND
AEF5: A2 04                  LDX  #$04.
                    RTFM
AEF7: 4C 03 AB               JMP  $AB03

--End assembly--

108 bytes

Errors: 0
```

```
1   REM  ***************************
2   REM  *                         *
3   REM  *    MREAD/MWRT INSTALL   *
4   REM  *                         *
5   REM  *    COPYRIGHT 1985 (C)   *
6   REM  *    BY COMPUTING AGE     *
7   REM  *                         *
8   REM  ***************************
100  D$ =  CHR$ (13) +  CHR$ (4)
110  PRINT D$"BLOAD MREAD/MWRT,A$A
   E8E"
120  POKE 40222,154: POKE 40223,17
   4: REM  ADDRESS -1 OF MWRT
130  POKE 40230,141: POKE 40231,17
   4: REM  ADDRESS -1 OF MREAD
140  FOR AD = 43140 TO 43143: READ
   BY: POKE AD,BY: NEXT : REM
   REPLACED INIT WITH MWRT
150  FOR AD = 43155 TO 43159: READ
   BY: POKE AD,BY: NEXT : REM  R
   EPLACE CHAIN WITH MREAD
160  POKE 43273,32: POKE 43274,123
   : REM  VALID KEYWORD FOR MWRT
   - A & L REQ'D ,S ,D,V & B OPT
   IONAL
170  POKE 43281,32: POKE 43282,123
   : REM VALID KEYWORD FOR MREAD
   X A & L REQ'D. S,D, V& B OPT
   IONAL
175  POKE 43773,76: POKE 43774,237
   : POKE 43775,174
180  HOME :A$ = "MREAD AND MWRT NO
   W INSTALLED": HTAB 20 -  LEN
   (A$) / 2: PRINT A$
190  DATA  77,87,82,212,77,82,69,6
   5,196
```

機械語言入門 吳博士

機械語言未來的命運

各位讀者，機械語言專欄至今已連載了5期。在今期我預備和各位檢討一下我們所認識的組合語言／機械語言與未來的趨勢發展。

我在這裏告訴各位，我們預期未來的發展趨勢會有利於高階語言，其理由是高階程式似乎總是會增加一些額外的性能，並且愈來愈大，例如AZTEC—C COMPILER 等的出現。第二是硬體與記憶體愈來愈便宜，而軟體與程式設計費用不斷增高。

單位位元成本低廉的大型記憶晶片不斷上市，例如68016（將來取代6502的16 BIT 6502版本）及（68032，68020等全32 BIT晶片），不太可能而且沒有必要節省這些晶片。加上更適合，更有效率的高階語言正在發展中，在很快的未來它們便會陸續出現而且勢必出現更標準化的高階語言。

微電腦的組合語言程式設計是不會像大型計算的組合語言一樣成爲乏人問津的藝術。但不過，因爲大程式，便宜的記憶體與昂貴的程式發展費用會使得軟體成本在應用上佔了大部份的比例，因此在此種情況下，許多的應用會趨於高階語言。

假如未來是高階語言的天下，爲甚麽我們連續了這麽多期的連載而且在以後還要作更深入的探討呢？

理由是目前很多的微電腦使用者仍用組合語言編寫程式，而且他們仍會繼續使用組合語言編寫程式，這是因爲使用者需要利用到組合語言所提供的有關控制方面的特殊功能。

另一方面是目前還沒有通用或標準化的高階語言發展出來，只有MBASIC是較合乎MACHINE INDEPENDENT的條件，而許多應用上我們可以利用組合語言所具有的高效率性。了解組合語言會有助於對高階語言的評估。

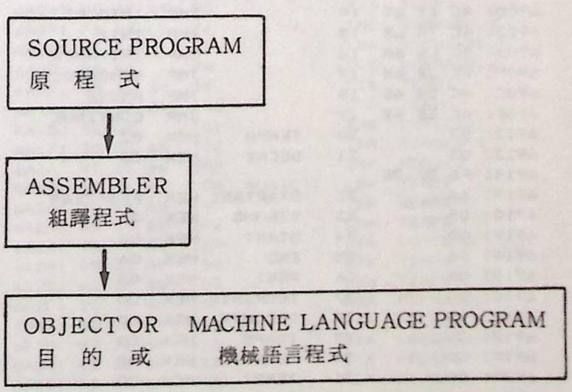
我們要儘量將組合語言編寫的程式合符下列條件：

1. 程式不會太大
2. 即時（REAL－TIME）的控制應用上
3. 有限度的資料處理
4. 大量生產的應用程式上
5. 輸入／輸出或控制動作次數遠超於計算動作

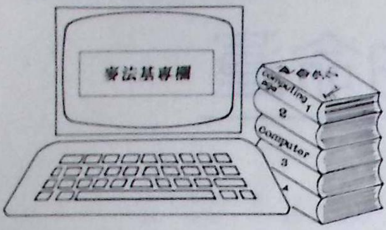
假如在你的應用中，硬體的成本高或速度是關鍵因素，那麽你最好使用組合語言，但這樣你必須準備花額外的時間在軟體發展上，以換取降低記憶體的成本與提高執行速度，反過來說如果在你的應用中，軟體佔了成本的大部份，那麽最好採用高階語言，不過你要準備額外的費用於硬件與軟體的支援方面。

當然，除了理論家之外無人會反對你同時使用組合語言及高階語言，開始時，你可以用高階語言來編寫程式，然後把某部份修改爲組合語言。然而大多數的使用者都避免這種作法，因爲這樣會使查錯，測試，以至說明程序的工作增加困難。

在以往我提到組合語言及機械語言，它們令很多讀者產生迷惑。事實上電腦只認識二進位數或十六位數的指令，例如＄20是機械碼，JSR是組合語言的助記憶碼，我們可以一個指令接一個指令地用人工方法來翻譯組合語言至電腦認識的機械碼，就像我們以往機械專欄一般，此種方法我們稱爲人工組合（HAND ASSEMBLY），不過組合是一種機械式的工作，我們可以將此工作交由微電腦來處理。微電腦在譯碼時從不發生錯誤，它總是知道每條指令的格式與所需要的語句。執行此工作的程式，我們稱爲組譯程式（ASSEMB—LER）如6502的LISA，MERLIN，BIG MAC，SC ASSEMBLER等，組譯程式將用助憶符號所編寫的使用者程式（USER PROGRAM）或是原程式（SOURCE PRO—GRAM）翻譯爲微電腦所能瞭解的機械語言程式（MAC—HINE LANGUAGE PROGRAM），或稱之爲目的程式（OBJECT PROGRAM）。組譯程式的輸入是原始程式，而輸出則是目的程式如下圖所示：



若果各位對本文花一些閱讀時間之後，你會確立自已編寫組合語言或其他高階語言的範疇，目的及軟件的價值觀，更重要的是不再失落！■



利用MOCKINGBOARD和MUSIC CONSTRUTION SET

爲你的程式配上背景音樂

相信大部份擁有MOCKINGBOARD的朋友都有MUSIC CONSTRUCTION SET 這個程式。由於操作簡單易用，所以甚受歡迎，只要按五綫譜把音樂符號塡上，就算對音樂不大熟識的朋友都可以獲得美妙音樂。

這些美妙音樂，原則上都可以加進任何程式之內，使我們一面應用程式而一面有背景音樂作爲陪襯。這裏就提出一種把音樂加進程式內的方法。

首先利用MUSIC CONSTRUCTION SET寫上一首自己喜愛的音樂，然後鍵入SAVE 指令把音樂存放在磁碟上。SAVE 指令只能把音樂檔案在MUSIC CONSTRUCTION SET的MASTER DISK 上使用，同時亦可以作爲以後修改之用，這個檔案並不能隨意自己運用。要獲取自己可以運用的檔案，必須在SAVE之後再鍵入BSAVE指令以另外一個檔案名稱存放磁碟上加以應用，這個檔案可以存放在任何所需的磁碟上。

BSAVE 之後的檔案是整首樂曲的資料，通常資料須要存放在高解像度第二頁的起始位置，不過大部份的程式都有可能應用第二頁或侵佔到第二頁以上的記憶體，所以資料存放位置亦要隨着改變。

參看LISTING＃1 的原始碼，這是產生音樂的程式，音樂資料檔案須要由這個程式負責產生樂音，我們可以稱之爲MUSIC 程式。在MUSIC CONSTRUCTION SET 內它是存放在由＄4900開始的位置上，這亦是一個不甚實際的位址，因爲同樣影响高解像第二頁，所以筆者把它改在＄6900上運行，這樣大部份的程式均可以應用了。讀者如果想改在其它位址上運行亦可，不過記着要把第248和250行的＄3FE和＄3FF一對指位器跟着更改，假設程式在＄6510上運行，則第247行須改爲＄10而第249行改爲＄65，以低高位元方式存入＄3FE和＄3FF內。

第22行的STARTADD亦要隨着資料檔案位置加以改變

```
LISTING #1         1
                   2
:ASM               3                ORG   $6900
                   4    ORB         EQU   $C400
                   5    ORA         EQU   $C401
                   6    ORB2        EQU   $C480
                   7    ORA2        EQU   $C481
                   8    DDRB2       EQU   $C482
                   9    DDRA2       EQU   $C483
                   10   DDRB        EQU   $C402
                   11   DDRA        EQU   $C403
                   12   SONGADD     EQU   $4
                   13   CHANNADD    EQU   $D6
6900: 4C B1 69     14               JMP   INTRUPT
6903: 4C 7C 6B     15               JMP   INIT
6906: 4C C8 6A     16               JMP   INITMOCK
6909: 4C 3D 6B     17               JMP   SONGADDS
690C: 4C D8 6B     18               JMP   PAUSE
690F: 4C EB 6B     19               JMP   CONTINUE
6912: 03           20   TEMPO       HEX   03
6913: 03           21   DECAY       HEX   03
6914: FE 5F 7E
6917: 64           22   STARTADD    HEX   FE5F7E64
6918: 0A           23   STRONG      HEX   0A
6919: 00           24   START       HEX   00
691A: 06           25   END         HEX   06
691B: 0A           26   REST        HEX   0A
691C: 00           27   TEMPCNTR    HEX   00
691D: 00           28   DECCNTR     HEX   00
691E: 00           29   TEMP5       HEX   00
691F: 00           30   TEMP0       HEX   00
6920: 00           31   TEMP1       HEX   00
6921: 00           32   TEMP2       HEX   00
6922: 00           33   TEMP4       HEX   00
6923: 00           34   TEMP6       HEX   00
6924: 01 01        35   CNTR        HEX   0101
6926: FF FF FF
6929: FF FF FF     36   VOICE       HEX   FFFFFFFFFFFF
692C: 00           37   PLUG        HEX   00
692D: 00           38   FREQLO      HEX   00
692E: 00           39   FREQHI      HEX   00
692F: 00 00        40   TIED        HEX   0000
6931: 1E 1F 20
6934: 22 24 26
6937: 29 2C        41   FREQLOS     HEX   1E1F20222426292C
6939: 2E 30 33
693C: 36 3A 3D
693F: 41 45        42               HEX   2E3033363A3D4145
6941: 49 4D 52
6944: 56 5C 61
6947: 67 6D        43               HEX   494D52565C61676D
6949: 73 7A 81
694C: 89 91 9A
694F: A3 AD        44               HEX   737A8189919AA3AD
6951: B7 C2 CE
6954: DA E7 F4
6957: 03 12        45               HEX   B7C2CEDAE7F40312
6959: 23 34 46
695C: 5A 6E 84
695F: 9B B3        46               HEX   2334465A6E849BB3
6961: CD E9 06
6964: 25 45 68
6967: 8C B3        47               HEX   CDE9062545688CB3
6969: DC 08 36
696C: 67 9B D2
696F: 01 01        48               HEX   DC0836679BD20101
6971: 00 00 00
6974: 00 00 00
6977: 00 00        49   FREQHIS     HEX   0000000000000000
6979: 00 00 00
697C: 00 00 00
697F: 00 00        50               HEX   0000000000000000
6981: 00 00 00
6984: 00 00 00
6987: 00 00        51               HEX   0000000000000000
6989: 00 00 00
```

，筆者把音樂資料存入＄7000起始位置上，所以STARTADD是FE6F7E74，如果改在＄6000上存放音樂資料，則須改爲FE5F7E64便可。

音樂的聲量亦可以由MUSIC 控制，一般來說，背景音樂不須要太响的音量。第23行的STRONG 值是音量的大小所在，數值越大則音量越大，相反來說便使音量減小。再看看第20行的TEMPO，這是控制音樂的快慢，大的數值會使音樂減慢，而第21行的DECAY 則控制樂音消逝時間，通常數值都不能大過TEMPO。第26行的REST是控制音色之用，數值小的話則樂音比較接近鋼琴的音色，而數值大的話則較接近電子琴的音色。

由於MUSIC 要利用＄300開始的第三頁記憶體作爲資料與MOCKINGBOARD溝通之用，所以任何程式利用第三頁都須要改到其他記憶位置上使用。最有可能的是一般BA－SIC 程式都利用第三頁作爲聲音副程式之用，當然亦要移到其他地方去。事實上，利用MOCKINGBOARD產生聲比較更加豐富，相信亦無須再用APPLE SPEAKER了。

一般編寫BASIC 程式的讀者通常都不會理會零頁區的問題，不過在這裏要留意的是MUSIC 會利用零頁區的＄4，＄5，＄6，＄7和＄D 6等，這些記憶位置千萬不能用。

最後，筆者利用MUSIC把一個名爲LION HUNT 的BASIC 遊戲更改成爲一個有背景音樂的遊戲，而且加上一MOCKINGBOARD的聲音使遊戲富於生氣，讀者可以作爲一個例子參考。這樣，所有BASIC 編寫的遊戲都不再是一些簡單而沒有變化的遊戲了。

LION HUNT原載於電腦時代第九期，這是一個十分出色的遊戲，玩法在遊戲內有說明，下面是一些加上的部份：

CALL 36952 （＄9058）

CALL 36980 （＄9074）

產生MOCKINGBOARD聲音

CALL 36897 （＄9021）

CALL 36941 （＄904D）

停止MOCKINGBOARD聲音

CALL 26883 （＄6903）

背景音樂產生

CALL 27608 （＄6BD8）

停止背景音樂

CALL 27627 （＄6BEB）

繼續背景音樂

A＝33024 （＄8100）

MOCKINGBOARD聲音表

第1794行的8，9，10和11是聲道指位器。

讀者如果未能編寫自己的遊戲，亦可以把其他人編寫的BASIC 遊戲更改爲MOCKINGBOARD產生聲音和音樂，相信會對APPLE 更加有興趣吧！

```
698C: 00 00 00
698F: 00 00      52            HEX  0000000000000000
6991: 00 00 00
6994: 00 00 00
6997: 01 01      53            HEX  0000000000000101
6999: 01 01 01
699C: 01 01 01
699F: 01 01      54            HEX  0101010101010101
69A1: 01 01 02
69A4: 02 02 02
69A7: 02 02      55            HEX  0101020202020202
69A9: 02 03 03
69AC: 03 03 03
69AF: 00 00      56            HEX  0203030303030000
69B1: 8A         57  INTRUPT   TXA
69B2: 48         58            PHA
69B3: 98         59            TYA
69B4: 48         60            PHA
69B5: A9 C0      61            LDA  #%11000000
69B7: 8D 0D C4   62            STA  $C40D
69BA: EE 1D 69   63  CHANCE    INC  DECCNTR
69BD: EE 1C 69   64            INC  TEMPCNTR
69C0: AD 1D 69   65            LDA  DECCNTR
69C3: CD 13 69   66            CMP  DECAY
69C6: D0 08      67            BNE  NODECAY
69C8: 20 A0 6B   68            JSR  DIMINISH
69CB: A9 00      69            LDA  #0
69CD: 8D 1D 69   70            STA  DECCNTR
69D0: AD 1C 69   71  NODECAY   LDA  TEMPCNTR
69D3: CD 12 69   72            CMP  TEMPO
69D6: F0 03      73            BEQ  MUSIC
69D8: 4C 6F 6A   74            JMP  MUSICRTI
69DB: A9 00      75  MUSIC     LDA  #$0
69DD: 8D 1C 69   76            STA  TEMPCNTR
69E0: A2 00      77            LDX  #0
69E2: 8E 21 69   78  MUSIC12   STX  TEMP2
69E5: 8A         79            TXA
69E6: 0A         80            ASL
69E7: 8D 22 69   81            STA  TEMP4
69EA: DE 24 69   82            DEC  CNTR,X
69ED: BD 24 69   83            LDA  CNTR,X
69F0: D0 6F      84            BNE  MUSIC9
69F2: 20 1E 6B   85            JSR  STOPVOIC
69F5: AE 22 69   86  MUSIC15   LDX  TEMP4
69F8: B5 04      87            LDA  SONGADD,X
69FA: 18         88            CLC
69FB: 69 02      89            ADC  #2
69FD: 95 04      90            STA  SONGADD,X
69FF: 90 02      91            BCC  MUSIC10
6A01: F6 05      92            INC  SONGADD+1,X
6A03: A1 04      93  MUSIC10   LDA  (SONGADD,X)
6A05: 4A         94            LSR
6A06: 8D 1F 69   95            STA  TEMPO
6A09: F6 04      96            INC  SONGADD,X
6A0B: A1 04      97            LDA  (SONGADD,X)
6A0D: D6 04      98            DEC  SONGADD,X
6A0F: 8D 20 69   99            STA  TEMP1
6A12: 0D 1F 69   100           ORA  TEMPO
6A15: D0 06      101           BNE  MUSIC11
6A17: 20 3D 6B   102           JSR  SONGADDS
6A1A: 4C 6C 6A   103           JMP  MUSIC3
6A1D: AE 21 69   104 MUSIC11   LDX  TEMP2
6A20: 20 76 6A   105           JSR  SEARCH
6A23: AD 21 69   106           LDA  TEMP2
6A26: 9D 26 69   107           STA  VOICE,X
6A29: 8E 23 69   108           STX  TEMP6
6A2C: AE 1F 69   109           LDX  TEMPO
6A2F: BD 31 69   110           LDA  FREQLOS,X
6A32: 8D 2D 69   111           STA  FREQLO
6A35: BD 71 69   112           LDA  FREQHIS,X
6A38: 8D 2E 69   113           STA  FREQHI
6A3B: AD 18 69   114           LDA  STRONG
6A3E: 8D 2C 69   115           STA  PLUG
6A41: AE 23 69   116           LDX  TEMP6
6A44: 20 9A 6A   117           JSR  PLUGIT
6A47: AE 21 69   118 MUSIC7    LDX  TEMP2
6A4A: AD 20 69   119           LDA  TEMP1
6A4D: 29 40      120           AND  #$40
```

編者按：鑑於LION HUNT　程式列表早已於第九期電腦時代刊載，作者只是作了輕微的修改，故不擬再在此刊登這遊戲的程式列表。本文只列出LISTING 1的 MUSIC主程式的SOURCE CODE。讀者將可以在本期的程式磁碟找到6個程式檔案，它們分別是：

＃1　MUSIC.S
＃2　MUSIC
＃3　LION HUNT
＃4　SHAPES 1
＃5　BLACKSMITH
＃6　MOCKINGBOARD

6個檔案之中，MUSIC.S是原始碼，而MUSIC是AS－SEMBLE後的OBJECT CODE檔案。

LION HUNT即是遊戲的主程式，而SHAPES 1是SHAPE TABLE。

BLACKSMITH是遊戲程式音樂背景DATA。而MOC－KINGBOARD檔案則是新加進的聲音效果DATA。

擁有MOCKING BOARD　的讀者，好好的利用它吧！

```
6A4F: 9D 2F 69             STA  TIED,X
6A52: AD 20 69             LDA  TEMP1
6A55: 29 3F                AND  #$3F
6A57: 9D 24 69             STA  CNTR,X
6A5A: AD 20 69             LDA  TEMP1
6A5D: 29 80                AND  #$80
6A5F: D0 94                BNE  MUSIC15
6A61: AE 21 69   MUSIC9    LDX  TEMP2
6A64: E8                   INX
6A65: E0 02                CPX  #$2
6A67: F0 03                BEQ  MUSIC3
6A69: 4C E2 69             JMP  MUSIC12
6A6C: 20 C8 6A   MUSIC3    JSR  INITMOCK
6A6F: 68         MUSICRTI  PLA
6A70: A8                   TAY
6A71: 68                   PLA
6A72: AA                   TAX
6A73: A5 45                LDA  $45
6A75: 40                   RTI
6A76: E0 00      SEARCH    CPX  #$0
6A78: D0 10                BNE  S2
6A7A: AE 19 69             LDX  START
6A7D: BD 26 69   S4        LDA  VOICE,X
6A80: 30 17                BMI  S3
6A82: E8                   INX
6A83: EC 1A 69             CPX  END
6A86: D0 F5                BNE  S4
6A88: CA                   DEX
6A89: 60                   RTS
6A8A: AE 1A 69   S2        LDX  END
6A8D: CA                   DEX
6A8E: BD 26 69   S5        LDA  VOICE,X
6A91: 30 06                BMI  S3
6A93: CA                   DEX
6A94: EC 19 69             CPX  START
6A97: D0 F5                BNE  S5
6A99: 60         S3        RTS
6A9A: 8A         PLUGIT    TXA
6A9B: 48                   PHA
6A9C: 98                   TYA
6A9D: 48                   PHA
6A9E: A9 00                LDA  #0
6AA0: E0 03                CPX  #3
6AA2: 90 05                BCC  PI2
6AA4: CA                   DEX
6AA5: CA                   DEX
6AA6: CA                   DEX
6AA7: A9 10                LDA  #$10
6AA9: 85 D6      PI2       STA  CHANNADD
6AAB: 8A                   TXA
6AAC: 0A                   ASL
6AAD: A8                   TAY
6AAE: AD 2D 69             LDA  FREQLO
6AB1: 91 D6                STA  (CHANNADD)
6AB3: C8                   INY
6AB4: AD 2E 69             LDA  FREQHI
6AB7: 91 D6                STA  (CHANNADD)
6AB9: 8A                   TXA
6ABA: 18                   CLC
6ABB: 69 08                ADC  #$8
6ABD: A8                   TAY
6ABE: AD 2C 69             LDA  PLUG
6AC1: 91 D6                STA  (CHANNADD),Y
6AC3: 68                   PLA
6AC4: A8                   TAY
6AC5: 68                   PLA
6AC6: AA                   TAX
6AC7: 60                   RTS
6AC8: 98         INITMOCK  TYA
6AC9: 48                   PHA
6ACA: A9 FF                LDA  #$FF
6ACC: 8D 03 C4             STA  DDRA
6ACF: 8D 83 C4             STA  DDRA2
6AD2: A9 07                LDA  #$7
6AD4: 8D 02 C4             STA  DDRB
6AD7: 8D 82 C4             STA  DDRB2
6ADA: A0 00                LDY  #$0
6ADC: 8C 01 C4   IM2       STY  ORA
6ADF: A9 07                LDA  #$7
6AE1: 8D 00 C4             STA  ORB
6AE4: A9 04                LDA  #$4
6AE6: 8D 00 C4             STA  ORB
6AE9: B9 00 03             LDA  $300,Y
6AEC: 8D 01 C4             STA  ORA
6AEF: A9 06                LDA  #$6
6AF1: 8D 00 C4             STA  ORB
6AF4: A9 04                LDA  #$4
6AF6: 8D 00 C4             STA  ORB
6AF9: 8C 81 C4             STY  ORA2
6AFC: A9 07                LDA  #$7
6AFE: 8D 80 C4             STA  ORB2
6B01: A9 04                LDA  #$4
6B03: 8D 80 C4             STA  ORB2
6B06: B9 10 03             LDA  $310,Y
6B09: 8D 81 C4             STA  ORA2
6B0C: A9 06                LDA  #$6
6B0E: 8D 80 C4             STA  ORB2
6B11: A9 04                LDA  #$4
6B13: 8D 80 C4             STA  ORB2
6B16: C8                   INY
6B17: C0 0F                CPY  #$F
6B19: D0 C1                BNE  IM2
6B1B: 68                   PLA
6B1C: A8                   TAY
6B1D: 60                   RTS
6B1E: AE 19 69   STOPVOIC  LDX  START
6B21: BD 26 69   SV2       LDA  VOICE,X
6B24: CD 21 69             CMP  TEMP2
6B27: D0 0D                BNE  SV3
6B29: A9 FF                LDA  #$FF
6B2B: 9D 26 69             STA  VOICE,X
6B2E: A9 00                LDA  #$0
6B30: 8D 2C 69             STA  PLUG
6B33: 20 9A 6A             JSR  PLUGIT
6B36: E8         SV3       INX
6B37: EC 1A 69             CPX  END
6B3A: D0 E5                BNE  SV2
6B3C: 60                   RTS
6B3D: AD 14 69   SONGADDS  LDA  STARTADD
6B40: 85 04                STA  SONGADD
```

爲你的程式配上背景音樂

```
6B42: AD 15 69             LDA  STARTADD+1
6B45: 85 05                STA  SONGADD+1
6B47: AD 16 69             LDA  STARTADD+2
6B4A: 85 06                STA  SONGADD+2
6B4C: AD 17 69             LDA  STARTADD+3
6B4F: 85 07                STA  SONGADD+3
6B51: A9 00                LDA  #$0
6B53: 8D FE 03             STA  $3FE
6B56: A9 69                LDA  #$69
6B58: 8D FF 03             STA  $3FF
6B5B: A9 00                LDA  #$0
6B5D: 85 D6                STA  CHANNADD
6B5F: A9 03                LDA  #$3
6B61: 85 D7                STA  CHANNADD+1
6B63: A9 01                LDA  #$1
6B65: 8D 24 69             STA  CNTR
6B68: 8D 25 69             STA  CNTR+1
6B6B: A9 00                LDA  #$0
6B6D: 8D 21 69             STA  TEMP2
6B70: 20 1E 6B             JSR  STOPVOIC
6B73: A9 01                LDA  #$1
6B75: 8D 21 69             STA  TEMP2
6B78: 20 1E 6B             JSR  STOPVOIC
6B7B: 60                   RTS
6B7C: 20 3D 6B   INIT      JSR  SONGADDS
6B7F: A9 F8                LDA  #$F8
6B81: 8D 07 03             STA  $307
6B84: 8D 17 03             STA  $317
6B87: A9 40                LDA  #%01000000
6B89: 8D 0B C4             STA  $C40B
6B8C: A9 C0                LDA  #%11000000
6B8E: 8D 0D C4             STA  $C40D
6B91: 8D 0E C4             STA  $C40E
6B94: A9 FF                LDA  #$FF
6B96: 8D 04 C4             STA  $C404
6B99: A9 40                LDA  #$40
6B9B: 8D 05 C4             STA  $C405
6B9E: 58                   CLI
6B9F: 60                   RTS
6BA0: A2 00      DIMINISH  LDX  #$00
6BA2: BD 26 69   D1        LDA  VOICE,X
6BA5: C9 02                CMP  #$2
6BA7: B0 11                BCS  D2
6BA9: A8                   TAY
6BAA: B9 2F 69             LDA  TIED,Y
6BAD: D0 0B                BNE  D2
6BAF: BD 08 03             LDA  $308,X
6BB2: CD 1B 69             CMP  REST
6BB5: F0 03                BEQ  D2
6BB7: DE 08 03             DEC  $308,X
6BBA: BD 29 69   D2        LDA  VOICE+3,X
6BBD: C9 02                CMP  #$2
6BBF: B0 11                BCS  D3
6BC1: A8                   TAY
6BC2: B9 2F 69             LDA  TIED,Y
6BC5: D0 0B                BNE  D3
6BC7: BD 18 03             LDA  $318,X
6BCA: CD 1B 69             CMP  REST
6BCD: F0 03                BEQ  D3
6BCF: DE 18 03             DEC  $318,X
6BD2: E8         D3        INX
6BD3: E0 03                CPX  #$3
6BD5: D0 CB                BNE  D1
6BD7: 60                   RTS
6BD8: 78         PAUSE     SEI
6BD9: A9 00                LDA  #$00
6BDB: 8D 21 69             STA  TEMP2
6BDE: 20 1E 6B             JSR  STOPVOIC
6BE1: EE 21 69             INC  TEMP2
6BE4: 20 1E 6B             JSR  STOPVOIC
6BE7: 20 C8 6A             JSR  INITMOCK
6BEA: 60                   RTS
6BEB: 58         CONTINUE  CLI
6BEC: 60                   RTS

--End assembly--
```

BLOAD位址指標

David

你也許從CATALOG 中看見一精采的B型檔，於是立刻BLOAD它，再BSAVE它。糟了！不知道檔案的起始點及長度怎BSAVE呢！也許你可把長度當作最大的假設，但這會浪費很多的SECTOR，而且起始點是無法估計的。你也可在＄AA60－＄AA74 之間找到最後處理的一個B型檔的長度和起始點，但這樣會容易出錯和緩慢。故這裏介紹一個程式，只要BRUN它，跟着BL－OAD你欲BSAVE的檔，再CALL－151 CTRL－Y RETURN，將會出現A＄nnnn，L＄nnnn（n是十六進位

```
95B5- A9 B5 85
95B8- 6F 85 73 A9 95 85 70 85
95C0- 74 A9 CC 8D F9 03 A9 95
95C8- 8D FA 03 60 A9 C1 20 ED
95D0- FD A9 A4 20 ED FD AD 73
95D8- AA 20 DA FD AD 72 AA 20
95E0- DA FD A9 AC 20 ED FD A9
95E8- CC 20 ED FD A9 A4 20 ED
95F0- FD AD 61 AA 20 DA FD AD
95F8- 60 AA 20 DA FD 4C 69 FF
```

值），這個程式可以包含在HELLO 程式內，使用後只是損失一部份的STRI－NG 貯存位置，因爲HIMEM 位值是被移低了一點，這個是不會由其他的－RUN 所重置的（程式剛好在DOS 之下＄95B5)，當然HIMEM 指令是另一個不同的指令，作別論計。機械語言內設置HIMEM 值是不同於蘋果BASIC的，它不會自動將STRING位置也一起更改，故此如果要和BASIC 同時使用，則要在程式前加上一HIMEM：38325的指令來保護，後會有期！

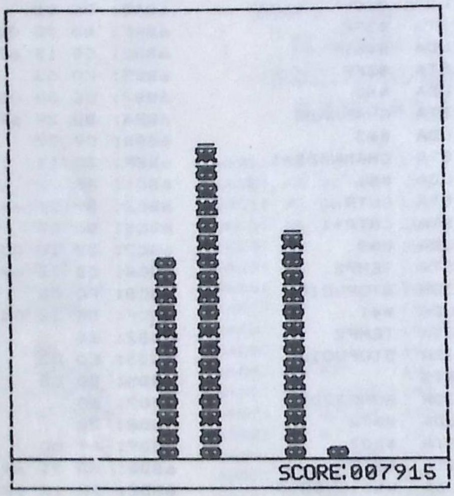
BSAVE BLOAD TRICKS，A＄95B5，L＄4B（BLOAD TRICKS是檔名）

一個與别不同的侵略者射擊遊戲

太空侵略者封閉戰 CLOSING ALIEN

大家有否厭倦於太空侵略者一類的射擊遊戲，老是按著FIRE 掣射擊永無休止襲來的敵人，有時會令你精神進入緊張狀態而不自覺。

筆者爲大家編寫了一個太空侵略者的射擊遊戲，但這不同於過去所有的侵略者遊戲。這個遊戲雖然是射擊侵略者，但却不是要你殲滅來侵者，而是要控制他們降落的位置，並將他們叠積得有如圖A所附的一樣。



遊戲玩法相當簡單，但要取得好成績必須要有良好的眼手配合。

要控制侵略者的移動，必須要射擊太空船兩邊，假如擊中太空船右邊，則它會向左飄移。假如射中太空船左邊，則它會飄移向右方。如果射中太空船正中心，則會隨機作左或右的移動。

得分的多寡視乎太空船降落的位置，最左及最右兩邊一行可得一百分。移入一行則得五十分，移入第二行則得廿五分，移入第三行可得十分。中間六行可取得五分。

在遊戲開始時，炮火可作整個畫面濶度的左右任意移動（利用左右箭咀鍵），但太空船降落後，或侵略者發射的炮火轟中地面，則己方炮火的移動會受到太空船所處的降落位置所限制。但無論如何，任何情況下，只可讓太空船降落在槍炮的左右，假如落正炮上，則遊戲會玩完。

至於射中侵略者發射的炮彈，就不會令它左右移動，而是會被擊毀，每擊中一枚可得150分。假如讓炮彈落在槍炮上，則遊戲亦會完結。每得一萬分，所有降落了的太空船會全部消失，而己方的槍炮亦可重新獲得最大的自由移動空間。

本遊戲共由兩個部份組成，一是一APPLESOFT 主程式，另一個是機械碼圖形表。

大家打完APPLESOFT 程式列表後，用下列指令SAVE入磁碟：

SAVE CLOSING ALIEN

至於機械碼圖形表，輸入法可參考第三期電腦時代，大家可用下列指令—SAVE入磁碟。

BSAVE ALIEN SHAPE，A＄6000，L＄21E。

如果大家想利用歡樂棒來進行遊戲，可更改APPLESOFT 程式內的下列數行：

```
200  HCOLOR= 3:X =  INT ( PDL
     (0) + (.054 * PDL (0)))
310  IF PEEK ( - 16287) ) 127 THEN
     GOSUB 470
1020 HCOLOR= 3:X =  INT ( PDL (0
     ) + (.054 * PDL (0)))
1100 IF PEEK ( - 16287) ) 127 THEN
     GOSUB 1190
```

```
****************
* ALIEN SHAPE *
* A$6000,L$21E*
****************
6000- 12 00 26 00 35 00 3E 00
6008- 4A 00 56 00 61 00 6D 00
6010- 79 00 81 00 8E 00 99 00
6018- B2 00 CB 00 D6 00 15 01
6020- 74 01 D2 01 05 02 29 2D
6028- 20 24 24 3B 3F 32 36 2E
6030- 28 28 20 00 00 2D 2D DC
6038- 24 24 BC 17 04 00 2D 2D
6040- DC 1B 64 2D 05 20 1C 3F
6048- 27 00 A8 2D 05 20 1C 67
6050- 21 1C 3F 17 04 00 49 24
6058- 05 38 3F 27 0C 0C 35 26
6060- 00 A8 2D.05 20 1C 3F 27
6068- 24 2D 2D 04 00 29 2D 20
6070- 1C 3F D6 24 0C 0C 0C 04
6078- 00 21 64 0C 0C 3C 3F 27
6080- 00 29 2D 20 1C 67 21 1C
6088- 3F 17 AE 1E 26 00 29 28
6090- 28 20 E4 3F 17 76 2D 04
6098- 00 2D 2D 2D 2D 25 3F 3F
60A0- 3F 3F 67 2D 2D 2D E5 3F
60A8- 3F 67 2D E5 27 25 27 25
60B0- 27 00 29 2D 2D 2D 05 38
60B8- 3F 3F 3F 3F 2C 4D 6D 29
60C0- 3C 3F 3F 3F 3F 0C 2D 2D
60C8- 2D 25 00 49 29 05 38 3F
60D0- 2C 2D 1C 27 25 00 3F BF
60D8- 76 2D 15 F6 3F 67 49 51
60E0- 29 2D F8 1B 24 24 0C 2D
60E8- 15 0D 36 36 0E 2D 05 20
60F0- 24 E4 3F 4C 49 32 36 36
60F8- 6E 09 1C 1C 07 28 2D 20
6100- 1C 3F 4C 49 32 36 36 2E
6108- 2D 25 C0 3F 04 80 28 2D
6110- 0D B6 12 26 00 07 38 3F
6118- 32 36 36 29 2D 20 01 08
6120- 18 08 30 36 36 2E 2D 2D
6128- 21 24 24 21 29 2A 32 3E
6130- 3F 0D 0A 2E 08 18 08 18
6138- 08 30 36 36 0E 2D 05 20
6140- 24 24 0D 09 09 3F 37 33
6148- 0E 2D 15 36 3B 3F 0F 09
6150- 09 09 18 08 18 08 18 08
6158- 2D 2D 1D 1B 32 36 36 0D
6160- 18 08 01 08 29 2D 20 04
6168- 3B 3F 36 36 36 0D 18 08
6170- 15 15 05 00 36 36 0E 2D
6178- 05 20 24 24 3B 3F 09 09
6180- 09 12 0A 2D 05 20 1C 3F
6188- 37 36 36 2E 09 09 21 24
6190- 24 2C 09 31 36 3F 37 09
6198- 31 2E 01 20 24 24 29 2D
61A0- 32 36 36 3B 3F 09 09 09
61A8- 24 24 24 2D 2D 32 1E 3F
61B0- 0F 11 09 36 3B 3F 09 09
61B8- 09 2D 1D 03 20 24 24 2F
61C0- 29 12 0A 12 22 24 24 21
61C8- 31 31 31 3E 3F 0E 09 2E
61D0- 00 05 36 36 2D 05 E0 27
61D8- 28 15 45 76 2D 20 95 DA
61E0- 37 6D 29 3C 04 20 35 0D
61E8- 08 36 36 4D 1C 07 28 05
61F0- F8 28 4D 36 76 2D 20 24
61F8- 0D 36 36 2D 05 20 FC 28
6200- 4D 76 36 04 00 31 1E 36
6208- 4D 09 2D 20 BF 23 64 2D
6210- 0D 36 76 2D 20 24 0D 76
6218- 36 25 2C 20 04 00 36
```

太空侵略者封閉戰

```
10  REM   **********************
20  REM   *   CLOSING ALIEN    *
30  REM   *   BY COMPUHOUSE    *
40  REM   * COPYRIGHT (C) 1985 *
50  REM   *  BY COMPUTING AGE  *
60  REM   *                    *
70  REM   **********************
80  LOMEM: 25600
90  ONERR  GOTO 1560
100  GOSUB 690: GOSUB 1410
110  GOSUB 1430: GOTO 130: REM
     INITIALIZE
120  CLEAR : GOSUB 1410
130 SC$ = "000000":SC = 0
140  DIM R(14),SC(13): FOR I = 0 TO
     13: READ SC(I): NEXT : DATA
     100,50,25,10,5,5,5,5,5,5,
     10,25,50,100
150  HGR2 : SCALE= 1: ROT= 0: HCOLOR=
     3:X = 130
160  HPLOT 0,0 TO 279,0 TO 279,18
     0 TO 0,180 TO 0,0: HPLOT 279
     ,180 TO 279,191 TO 0,191 TO
     0,180
180  DRAW 14 AT 203,182: GOSUB 76
     0
190 MI = 12:MA = 252:Y1 = 9
200  HCOLOR= 3:Z =  PEEK ( - 1638
     4): IF Z = 136 OR Z = 149 THEN
     POKE  - 16368,0:M =  ABS (X
     - X1):X =  INT (X + (5 * (M
     <  = 20) + .5 * (M > 20) *
     M) * ((Z = 149)  (Z = 136))
     )
210  IF MA - MI < 20 THEN 830
220  IF SC - RE >  = 10000 THEN R
     E = RE + 10000: FOR I = 0 TO
     13:R(I) = 0: NEXT : GOTO 150
230  IF Y1 = 9 AND  INT (( RND (1
     ) * 10) / 3) = ( INT ( RND (
     0) * 10)) / 3 THEN 990
240  IF X < MI THEN X = MI: GOTO
     260
250  IF X > (MA) THEN X = MA
260  XDRAW 11 AT X,179
270  IF Y1 > 9 THEN 290
280 R =  INT ( RND (1) * 14):X1 =
     (R * 20) + 2: IF X1 < MI OR
     X1 > (MA) THEN 280
290  GOSUB 380: REM   CHECK POSIT
     ION
300  DRAW 12 AT X1,Y1
310 Z =  PEEK ( - 16384): IF Z =
     160 THEN  POKE  - 16368,0: GOSUB
     470
320  GOSUB 410
330  HCOLOR= 0
340  IF Y1 <  = 179 THEN  DRAW 12
     AT X1,Y1:Y1 = Y1 + 5
350  IF Y1 > 169 THEN  GOSUB 860
360  IF Y1 = 179 OR Y1 = 179 - (R
     (R) * 5) THEN Y1 = 9:R(R) =
     R(R) + 1: GOSUB 710: GOSUB 4
     20
370  GOTO 200
380  REM  :CHECK FOR TOP OF COLUM
     N POSITION
390  IF Y1 = 179 -  (R(R) * 5) - 5
     THEN  HCOLOR= 3: XDRAW 12 AT
     X1,Y1 + 5:Y1 = 9:R(R) = R(R)
     + 1: GOSUB 410: GOSUB 710: GOSUB
     420: POP : GOTO 200
400  RETURN
410  XDRAW 11 AT X,179: RETURN
420  REM  :SET RANGE OF GUN
430  IF X1 = MI OR X1 = (MA) THEN
     RETURN
440  IF X1 < X AND X1 > MI THEN M
     I = X1 + 10: RETURN
450  IF X1 > X AND X1 < (MA) THEN
     MA = X1 - 10
460  RETURN
470  REM  :FIRE SHOT
480  HPLOT X + 5,169 TO X + 5,Y1:
     GOSUB 680: HCOLOR= 0: HPLOT
     X + 5,169 TO X + 5,Y1: HCOLOR=
     3
490  IF X = X1 OR X = X1 + 1 OR X
     = X1 - 1 THEN 560
500  IF X = X1 + 2 OR X = X1 + 3 OR
     X = X1 + 4 OR X = X1 + 5 THEN
     GOSUB 600:X1 = X1 - 20:R =
     R - 1: GOTO 530
510  IF X = X1 - 2 OR X = X1 - 3 OR
     X = X1  - 4 OR X = X1 - 5 THEN
     GOSUB 620:X1 = X1 + 20:R =
     R + 1: GOTO 530
520  RETURN
530  IF X1 < 2 THEN X1 = 2: RETURN
540  IF X1 > 262 THEN X1 = 262
550  RETURN
560  REM  :RANDOM MOVE FOR CENTER
     STRIKE
570 RM =  INT ( RND (1) * 2) + 1:
     IF RM / 2 =  INT (RM / 2) THEN
     590
580  GOSUB 620:X1 = X1 + 20:R = R
     + 1: GOSUB 380: GOTO 530
590  GOSUB 600:X1 = X1 - 20:R = R
     - 1: GOSUB 380: GOTO 530
600 TX = X1 - 20:TY = Y1 + 5: IF
     179 - (R(R - 1) * 5) <  = TY
     THEN  POP : POP : GOTO 320
610  HCOLOR= 0: DRAW 12 AT X1,Y1:
```

```
  HCOLOR= 3: RETURN
620  IF R = 13 THEN  POP : POP : GOTO
   320
630 TX = X1 + 20:TY = Y1 + 5: IF
   179 - (R(R + 1) * 5) < = TY
    THEN  POP : POP : GOTO 320
640  GOTO 610
650  REM  : SOUND OF EXPLOSION
660  FOR J = 1 TO 4: POKE  - 1633
   6, PEEK ( - 16336): NEXT : RETURN

670  REM  :SOUND OF SHOT
680  CALL 770: RETURN
690  POKE 768,124: POKE 769,10: POKE
   770,173:,POKE 771,48: POKE 7
   72,192: POKE 773,165: POKE 7
   74,0: POKE 775,233: POKE 776
   ,1: POKE 777,208: POKE 778,2
   52: POKE 779,230
700  POKE 780,0: POKE 781,208: POKE
   782,243: POKE 783,96: RETURN

710  REM  :UPDATE SCORE
720  HCOLOR= 0: FOR J = 1 TO 6:C =
   ASC ( MID$ (SC$,J,1)) - 47:
   ON J GOSUB 770,780,790,800,
   810,820: NEXT : HCOLOR= 3
730  IF FL THEN  RETURN
740 SC = SC + SC(R):SC$ =  STR$ (
   SC)
750  IF  LEN (SC$) < 6 THEN SC$ =
   "0" + SC$: GOTO 750
760  FOR J = 1 TO 6:C =  ASC ( MID$
   (SC$,J,1)) - 47: ON J GOSUB
   770,780,790,800,810,820: NEXT
   : RETURN
770  DRAW C AT 233,188: RETURN
780  DRAW C AT 240,188: RETURN
790  DRAW C AT 247,188: RETURN
800  DRAW C AT 254,188: RETURN
810  DRAW C AT 261,188: RETURN
820  DRAW C AT 268,188: RETURN
830  DRAW 11 AT MI,179
840  FOR I = 9 TO 179 STEP 5: DRAW
   12 AT MI,I: POKE  - 16336, PEEK
   ( - 16336): NEXT
850 X = MI: GOTO 880
860  IF X = X1 OR X = X1 + 1 OR X
   = X1 + 2 OR X = X1 + 3 OR X
   = X1 + 4 OR X = X1 + 5 OR X
   = X1 - 1 OR X = X1 - 2 OR X
   = X1 - 3 OR X = X1 - 4 OR X
   = X1 - 5 THEN 880
870  RETURN
880  HCOLOR= 3: FOR I = 1 TO 5: SCALE=
   I: GOSUB 650: DRAW 11 AT X,1
   79: NEXT
890  HCOLOR= 0: FOR I = 1 TO 5: SCALE=
   I: GOSUB 650: DRAW 11 AT X,1
   79: NEXT : SCALE= 1
900  FOR I = 1 TO 1000: NEXT : TEXT
   : HOME
910  IF SC < HS THEN 960
920  VTAB 12: FLASH : PRINT "NEW
   HIGH SCORE :"SC:HS = SC: NORMAL
   : PRINT
930  INPUT "PLEASE ENTER YOUR NAM
   E :";PL$: IF PL$ = "" THEN 9
   30
940  PRINT  CHR$ (4)"OPEN HIGH.SC
   ORE": PRINT  CHR$ (4)"WRITE
   HIGH.SCORE": PRINT HS: PRINT
   PL$: PRINT  CHR$ (4)"CLOSE"
950  GOTO 970
960  VTAB 12: PRINT "HIGH SCORE I
   S ";: FLASH : PRINT HS;: NORMAL
   : PRINT " BY "PL$"."
970  VTAB 22: INPUT "PLAY AGAIN ?
   ";YN$: IF  LEFT$ (YN$,1) = "
   Y" THEN  FOR I = 0 TO 14:R(I
   ) = 0: NEXT :SC = 0:SC$ = "0
   00000": GOTO 120
980  HOME : FOR I = 38 TO 1 STEP
    - 1: GOSUB 650: VTAB 24: HTAB
   I: PRINT  CHR$ (93)" ";: NEXT
   : VTAB 23: END
990  REM  :BOMB SEQUENCE
1000 B =  INT ( RND (1) * 3 + 1)
1010 S =  INT ( RND (1) * 5) * 10
   :S = Y1 + S:Y1 = S
1020  HCOLOR= 3:Z =  PEEK ( - 163
   84): IF Z = 136 OR Z = 149 THEN
    POKE  - 16368,0:M =  ABS (X
    - X1):X =  INT (X + (5 * (M
   < = 20) + .5 * (M > 20) *
   M) * ((Z = 149) - (Z = 136))
   )
1030  IF X < MI THEN X = MI
1040  IF X > (MA) THEN X = MA
1050  XDRAW 11 AT X,179
1060  IF Y1 > S THEN 1090
1070 R =  INT ( RND (1) * 14):X1 =
   (R * 20) + 2: IF X1 < MI OR
   X1 > (MA) THEN 1070
1080  FOR I = 1 TO B:B(I) = X1:X1
    = X1 + 5: NEXT
1090  FOR I = 1 TO B: DRAW 13 AT
   B(I),Y1: NEXT
1100 Z =  PEEK ( - 16384): IF Z =
   160 THEN  POKE  - 16368,0: GOSUB
   1190
1110  HCOLOR= 0
1120  FOR I = 1 TO B: DRAW 13 AT
   B(I),Y1: NEXT
1130  DRAW 11 AT X,179: FOR L = 1
   68 TO 170: HPLOT X + 3,L TO
```

```
 X + 6,L: NEXT : GOSUB 410:Y1
   = Y1 + 10
1140  FOR I = 1 TO B: IF Y1 = 179
   THEN  GOSUB 1360: IF B = I THEN
  BG = 1:Y1 = 9: GOSUB 410
1150  NEXT
1160  IF EG = 1 THEN EG = 0:BG =
  0: GOTO 880
1170  IF BG = 1 THEN BG = 0: GOTO
  200
1180  GOSUB 410: GOTO 1020
1190  REM  :SHOOT AT BOMBS
1200  HPLOT X + 5,169 TO X + 5,Y1
   + 1: GOSUB 680: HCOLOR= 0: HPLOT
  X + 5,169 TO X + 5,Y1 + 1: HCOLOR= 3
1210  FOR J = 1 TO B: IF X > B(J)
   - 2.5 AND X < B(J) + 2.5 THEN
  I = J:J = B:EX = 1
1220  NEXT
1230  IF EX = 1 THEN EX = 0: GOTO
  1250
1240  RETURN
1250  HCOLOR= 3: FOR K = 1 TO 3: SCALE= K:
  XDRAW 13 AT B(I),Y1: GOSUB 650
  : NEXT
1260  HCOLOR= 0: FOR K = 2 TO 3: SCALE= K:
  XDRAW 13 AT B(I.,Y1: GOSUB 650
  : NEXT : SCALE= 1
1270 FL = 1: GOSUB 710:FL = 0
1280 SC = SC + 150:SC$ =  STR$ (S
  C)
1290  IF  LEN (SC$) < 6 THEI' SC$ =
  "0" + SC$: GOTO 129C
1300  GOSUB 760
1310  IF I = B AND B = 1 THEN Y1 =
  9: HCOLOR= 0: DRAW 11 AT X,1
  7º: FOR L = 168 TO 170: HPLOT
  X + 3,L TO X + 6,L: NEXT : POP
  : GOTO 200
1320  IF I = B THEN B = B - 1: RETURN

1330  IF I = 1 AND B = 2 THEN B(I
  ) = B(I + 1):B = 1: RETURN
1340  IF I = 1 AND B = 3 THEN B(I
  ) = B(I + 1):B(I + 1) = B(I +
  2):B = 2: RETURN
1350  IF I = 2 AND B = 3 THEN B(I
  ) = B(I + 1):B = 2: RETURN
1360  IF X > B(I) - 2.5 AND X < B
  (I) + 2.5 THEN  HCOLOR= 3: FOR
  K = 1 TO 3: SCALE= K: XDRAW
  13 AT B(I),179: GOSUB 650: NEXT

1370  IF X > B(I) - 2.5 AND X < B
  (I) + 2.5 THEN  HCOLOR= 0: FOR
  K = 1 TO 3: SCALE= K: XDRAW
  13 AT B(I),179: GOSUB 650: NEXT
  : HCOLOR= 3: SCALE= 1:EG = 1
  : RETURN
1380  HCOLOR= 3: FOR K = 1 TO 3: SCALE=
  XDRAW 13 AT B(I),179: GOSUB 65
  0: NEXT : HCOLOR= 0: FOR K =
  1 TO 3: SCALE= K: XDRAW 13 AT
  B(I),179: GOSUB 650: NEXT : HCOLOR= 3:
  SCALE= 1
1390  IF B(I) / 20 <  >  INT (B(I
  ) / 20) THEN B(I) = B(I) - 1
  : GOTO 1390
1400 X1 = B(I): GOSUB 420: RETURN

1410  PRINT  CHR$ (4)"VERIFY HIGH
  .SCORE": PRINT  CHR$ (4)"OPE
  N HIGH.SCORE": PRINT  CHR$ (
  4)"READ HIGH.SCORE": INPUT H
  S,PL$: PRINT  CHR$ (4)"CLOSE
  "
1420  RETURN
1430  TEXT : HOME
1440  VTAB 12: HTAB 13: INVERSE :
  PRINT "CLOSING ALIEN ": PRINT
  : HTAB 12: PRINT "BY:  COMPU
  HOUSE ": PRINT : PRINT "** C
  OPYRIGHT 1985 BY  COMPUTING
  AGE  **": NORMAL
1450  IF HS > 0 THEN  PRINT : PRINT
  "PRESENT HIGH SCORE IS "HS"
  BY "PL$
1460  PRINT  CHR$ (4)"BLOAD ALIEN
  SHAPE"
1470  POKE 232,0: POKE 233,96
1480  VTAB 22: INPUT "DO YOU NEED
  INSTRUCTIONS ?";YN$: IF  LEFT$
  (YN$,1) = "N" THEN  RETURN
1490  HOME : HTAB 14: INVERSE : PRINT
  "CLOSING ALIEN ": NORMAL : PRINT

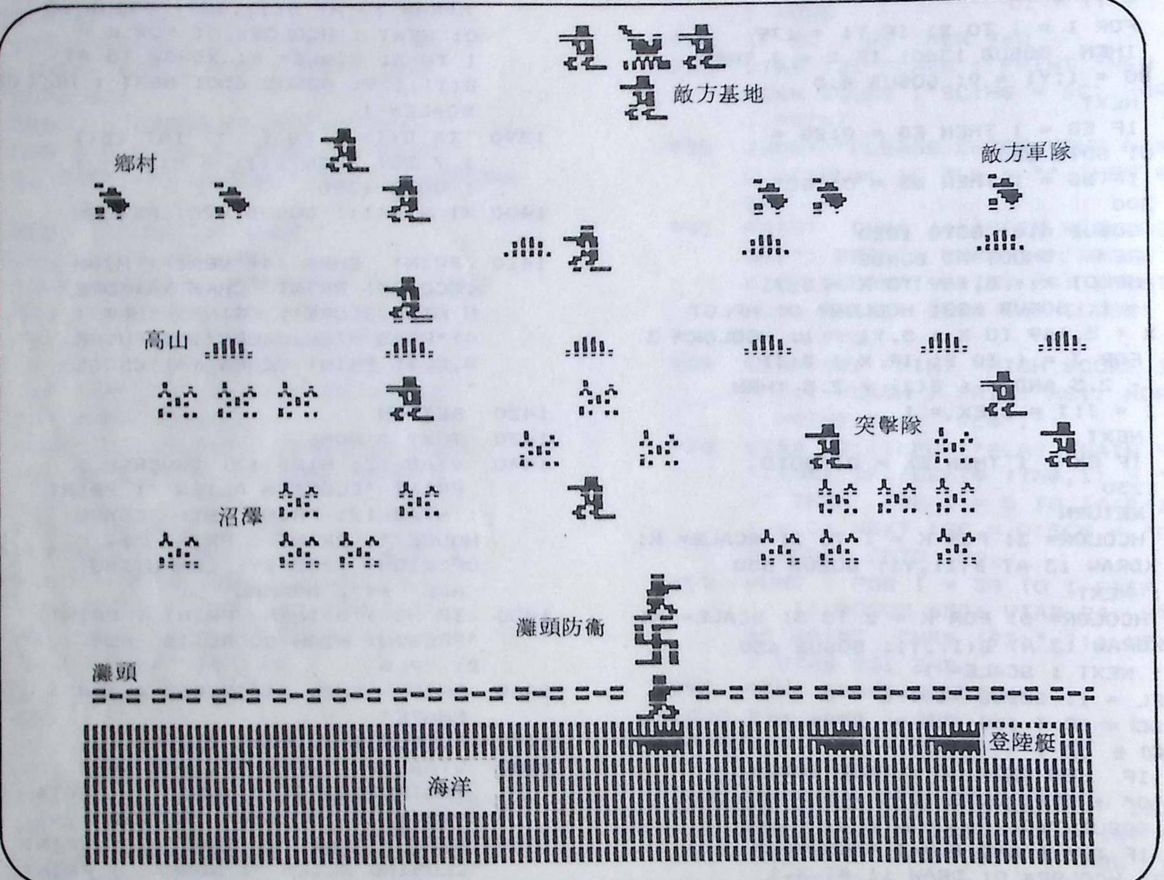
1500  VTAB 12: INVERSE : PRINT "P
  LEASE REFER TO THE MAGAZINE"
  : NORMAL
1540  VTAB 21: PRINT "PRESS ANY K
  EY TO CONTINUE";: GET YN$
1550  RETURN
1560  REM   : ERROR ROUTINE
1570 ER =  PEEK (222):EL =  PEEK
  (218) +  PEEK (219) * 256: POKE
  216,0
1580  IF ER = 6 THEN  PRINT  CHR$
  (4)"OPEN HIGH.SCORE": PRINT
  CHR$ (4)"WRITE HIGH.SCORE":
  PRINT 0: PRINT : PRINT  CHR$
  (4)"CLOSE": RUN
1590  IF ER = 255 THEN  TEXT : GOTO
  980
1600  TEXT : HOME : VTAB 12: PRINT
  "ERROR #"ER" HAS OCCURRED IN
  LINE "EL
1610  PRINT : PRINT "PLEASE REFER
  TO YOUR APPLESOFT MANUAL." ■
```

GRAPHICS MAGAICIAN編寫的遊戲：

動態／策略兼備的戰爭遊戲

BEACH HEAD

麥法基



電腦遊戲之中，戰棋遊戲是比較吸引的一種，由於富有挑戰性，所以筆者一直以來都希望編寫一個戰爭遊戲。這個BEACH HEAD基於以上原因編寫而成，不過因為時間不甚充份，是否有不盡之處希望讀者加以指正。然而遊戲已經盡量造到符合一般WAR GAME的要求，相信可以達到戰爭遊戲的目的。

這個程式對記憶體要求甚大，所以必須加上16K RAM CARD才能運行。最主要是把DOS搬上RAM CARD 使容量加大10K讓程式獲得充份記憶存放資料，所以磁碟上必須有第10期的RAM DOS存在，否則會HANG機。

遊戲比較複雜，讀者請小心閱讀本文對遊戲清楚如何進行才開始。遊戲概念是第二次大戰時德國在歐洲某處地方設立一個基地用以研究飛彈發射，而可靠消息知道該研究快將成功，這對盟軍來說是十分不利，而閣下被指派為一次行動的指揮官，帶領十二隊海軍突擊隊到該處進行破壞工作。每一隊突擊隊員均經過嚴格訓練來達成此項任務，不過該基地防守甚嚴，而且有空軍部隊部署，所以任務是否成功則在乎你的策略運用和戰鬥能力了。

遊戲分為幾個不同階段進行，現分述如下：

㈠登陸期（LANDING PHASE）

三艘登陸艇會在三處不同灘頭讓突擊隊員登陸，當遊戲開始時電腦會供給閣下每一隊突擊隊資料，包括SQUAD ♯突擊隊編號、CP（COMBAT POWER）戰鬥能力；數值越大則戰鬥力越高。MP（MOVEMENT POINT） 軍隊移動次數；FR（FATIGUE RATE）軍隊士氣，數值越大則士氣低落，TERRAIN 軍隊所置身地形，EFF（EFFICIENCY）戰鬥效率。閣下身為指揮官，當然需要對以上各點加以了解。

COMBAT POWER 戰鬥能力並非空中樓閣的一個數值，亦非單把戰鬥任務交由電腦計算。因為這個遊戲共分為兩大環節，當閣下一切策略部署完畢後，則當然會有作戰發生，這是戰爭無可避免的事。真正的作戰亦然由閣下所操作，這時螢幕會由地形分佈轉為作戰的ARCADE GAME畫面，閣下必須把敵人盡快擊退或消滅，而敵人亦有不同的戰鬥能

BEACH HEAD

力，很明顯地戰鬥能力是軍隊效率的一個重要因數，下面會再談。

MOVEMENT POINT 是軍隊移動次數，沒有一隊軍隊可以長時間移動，所以次數必有所限制，不過倘若軍隊在一次移動時停留下來，則該軍隊次數會保留在下一次移動時加上，所以移動次數是戰略上運用重要一環。

FATIGUE RATE軍隊士氣計算，雖然有良好的戰鬥能力，但士氣低落會影响效率，長時期行軍或作戰完畢皆影响士氣減低，必須要注意。

TERRAIN 地形環境亦是重要，地形分佈分爲以下幾個，依次序影响軍隊效率：

1 BEACH灘頭，2 SWAMP沼澤，3 VILLAGE鄉村，4 MOUNTAIN高山，5 OCEAN海洋。而CLEAR是上面所述以外地方，並不影响軍隊效率。

EFFICIENCY 軍隊效率由以上各個因數計算出來，所以數值與效率成正比例，這是要留意的地方。

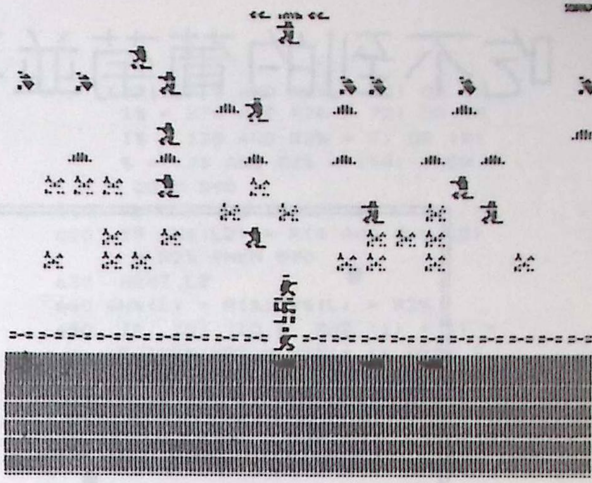
登陸期突擊隊會遇上稍弱的灘頭攻擊，電腦會供給閣下灘頭防衛力量的各種資料，而閣下須根據這些資料進行反擊，一個納粹圖形顯示灘頭的防衛所在，電腦會詢問你用那一種方式作戰（STRATEGY LEVE）數字1至9代表輕微攻擊至重創對方加以消滅，要小心保留實力，因爲數字大會影响下一次作戰效率，請小心作出决定。但突擊隊亦有可能被對方所消滅。

當所有突擊隊登陸後，這時敵方的空中部隊會立即起飛向突擊隊轟炸，畫面亦會改爲作戰部份，你必須在對方未投彈前把飛機擊落，否則傷亡會加大。

轟炸完畢則對方軍隊會移動，當然最終目的把突擊隊消滅。當敵人和突擊隊員相遇時作戰即會發生，畫面亦會轉爲ARCADE ，盡快把對方消滅，對方命中率由他的COMBAT POWER 所决定，而作戰時間則由大衆的效率所决定。換言之，對方效率越高你攻擊對方時間越短，因爲可能很快便會被對方消滅，而且對方擊中你的機會亦越大。

(二)突擊隊移動期

作戰完畢則改由你指揮隊員移動，移動時亦可選擇對敵人加以反擊，直至到達敵方基地爲止。基地亦會對你作出攻



擊行動，而且效率極高，所以要小心應付。

倘若把基地消滅，則任務完成。假如突擊隊員全數被消滅，則任務失敗。

遊戲會照以上規則循環進行，直至完畢爲止。

當進行這個遊戲時，策略是十分重要因數，要隨時留意隊員的效率，士氣和移動。盡量避免孤軍作戰，因爲就算一隊被消滅，另外一隊亦可以把對方反擊。移動時不要作無謂的攻擊，因爲任務是把對方基地消滅，所以要保留實力。不過倘若你把敵人一隊消滅後，基地的效率亦由於士氣而打上折扣。

原則上地形和軍隊分佈應該每次一樣，不過爲了增加遊戲的趣味，所以各種分佈形勢和軍隊實力每次不同，這樣每一次都有不同的遊戲進行。

祝各位好運！

本遊戲共由五個檔案組成。鑑於程式列表過於冗長，故此不在此登出。各位讀者可在本期出版的程式磁碟找到有關的檔案。

這幾個列表包括：

(1) BEACH HEAD
(2) BEACH. ANM
(3) BEACH. TXT
(4) BEACH. HEAD. PIC
(5) BEACH. OB

大家只要執行BEACH HEAD 便可。不過大家必須留意，進行本遊戲時，必須連同本刊第十期的RAM DOS 放在一起。■

我們接納任何形式的稿件。

投稿規則

(一) 本刊歡迎任何形式的投稿，包括軟件、硬件評介，及製作，工商業應用、遊戲軟件評介，電腦應用技巧，電腦程序、語言編寫技巧。

(二) 所有軟件程式投稿，必須要連同磁碟寄來或携來本出版社，程式必須附加清楚說明，用法及原理。交來之磁碟於文章發表後，即退回。如要退回原稿，先付回郵信封。

(三) 稿件長度字數不限，著重言之有物。

(四) 所有稿件必須附上作者中英文姓名，通訊電話及地址，身份證號碼，發表時筆名隨意。

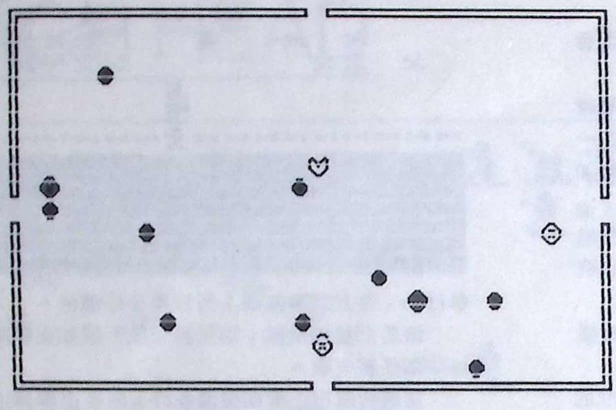
(五) 稿件必須用原稿紙橫排書寫，一字一格。

(六) 請勿一稿兩投。

(七) 稿酬於文章發表後即支持。

(八) 來稿請寄九龍中央郵局信箱71193 號電腦時代收，信封請勿黏口（有磁碟者除外），信封面請註明〔投稿〕，或親身携來本出版社。

吃不到的葡萄並不是酸的!!



這是一個高解像ARCADE 型式的遊戲。雖然是用BASIC 寫成，但反應並不慢。最大的特色是越玩下去，難度越高。

在這遊戲之中，你專門吃掉那些葡萄。不過有些葡萄可能會變成精靈，反過來追殺閣下。閣下可用A、Z 及左右箭咀鍵來逃避葡萄精靈，但並不表示不適宜吃掉，相反，如果能在一個難度中將所有變酸的葡萄吃掉，則可獲得額外的獎分。你可以利用4個出口來逃避葡萄精靈的追殺。■

```
2  REM  ****************
3  REM  ** SOUR GRAPE **
4  REM  **     BY     **
5  REM  ** COMPUHOUSE **
6  REM  **  (C) 1985  **
7  REM  ****************
10  GOSUB 710
20  GOSUB 530
30  FOR L = 1 TO NM%
40  XDRAW MT% AT MH%(L),MV%(L)
50  IF  INT (NG% *  RND (1) + 1) <
    = LE% THEN  GOTO 100
60  IF MH%(L) < HP% THEN MH%(L) =
   MH%(L) + IN%: GOTO 150
70  IF MH%(L) > HP% THEN MH%(L) =
   MH%(L) - IN%: GOTO 150
80  IF MV%(L) < VP% THEN MV%(L) =
   MV%(L) + IN%: GOTO 150
90  IF MV%(L) > VP% THEN MV%(L) =
   MV%(L) - IN%: GOTO 150
100 R1% =  INT (4 *  RND (1) + 1)

110  IF R1% = 1 AND MH%(L) > 9 THEN
   MH%(L) = MH%(L) - IN%: GOTO
   150
120  IF R1% = 2 AND MH%(L) < 270 THEN
   MH%(L) = MH%(L) + IN%: GOTO
   150
130  IF R1% = 3 AND MV%(L) > 9 THEN
   MV%(L) = MV%(L) - IN%: GOTO
   150
140  IF R1% = 4 AND MV%(L) < 144 THEN
   MV%(L) = MV%(L) + IN%
150  DRAW MT% AT MH%(L),MV%(L):CL
   % =  PEEK (234): IF CL% = 0 THEN
   200
160  IF MH%(L) = HP% AND MV%(L) =
   VP% THEN 450
170  IF CL% = 25 THEN  XDRAW MT% AT
   MH%(L),MV%(L): GOTO 200
180  IF CL% = 21 THEN  HCOLOR= 0:
    DRAW SG% AT MH%(L),MV%(L): DRAW
   MT% AT MH%(L),MV%(L): HCOLOR=
   3:GE% = GE% + 1: GOSUB 510: GOTO
   150
190  IF CL% = 5 THEN  HCOLOR= 0: DRAW
   GP% AT MH%(L),MV%(L): DRAW M
   T% AT MH%(L),MV%(L): HCOLOR=
   3:GE% = GE% + 1: GOSUB 510: GOTO
   150
200  NEXT
210  IF  PEEK ( - 16384) < 128 THEN
   30
220 KY% =  PEEK ( - 16384)
230  ROT= RT%: XDRAW GG% AT HP%,V
   P%
240  IF KY% = 149 AND HP% < 270 THEN
   RT% = 0:HP% = HP% + IN%: GOTO
   290
250  IF KY% = 193 AND VP% > 9 THEN
   RT% = 48:VP% = VP% - IN%: GOTO
   290
260  IF KY% = 136 AND HP% > 9 THEN
   RT% = 32:HP% = HP% - IN%: GOTO
   290
270  IF KY% = 218 AND VP% < 144 THEN
   RT% = 16:VP% = VP% + IN%: GOTO
   290
280  IF (KY% = 149 AND HP% = 270 AND
   VP% = 72) OR (KY% = 136 AND
   HP% = 9 AND VP% = 72) OR (KY
   % = 193 AND HP% = 135 AND VP
   % = 9) OR (KY% = 218 AND HP%
    = 135 AND VP% = 144) THEN 4
   00
290  ROT= RT%: DRAW GG% AT HP%,VP
   %: ROT= 0
300 CL% =  PEEK (234): IF CL% > 0
    THEN 320
310  GOTO 30
320  IF CL% = 15 THEN 450
330  HCOLOR= 0: ROT= RT%: DRAW GG
   % AT HP%,VP%: ROT= 0: DRAW G
   P% AT HP%,VP%: DRAW SG% AT H
   P%,VP%: HCOLOR= 3
340  IF CL% = 5 THEN  POKE 768,50
   : POKE 769,10: CALL 770:SC% =
   SC% + LE%:GE% = GE% + 1: GOSUB
   510: GOTO 290
```

吃不到的葡萄並不是酸的!!

```
350  IF CL% = 13 THEN  FOR L = 20
      TO 30: POKE 768,L: POKE 769
     ,2: CALL 770: NEXT L:SC% = S
     C% + (10 * LE%):GE% = GE% +
     1:SE% = SE% + 1: GOSUB 510: GOTO
     290
360  HOME : FLASH : VTAB 21: PRINT
     "LEVEL";LE%;" COMPLETED.": NORMAL

370  IF SE% = NS% AND NS% > 0 THEN
      VTAB 22: PRINT "ALL SOUR GR
     APES GOBBLED. BONUS: 10 X ";
     NS%:SC% = SC% + (10 * NS%)
380  FOR L = 1 TO 3: FOR L2 = 1 TO
     250 STEP 5: POKE 768,L2: POKE
     769,5: CALL 770: NEXT L2: NEXT
     L
390 LE% = LE% + 1: GOTO 20
400 R1% =  INT (4 *  RND (1) + 1)

410  IF R1% = 1 THEN RT% = 0:HP% =
     9:VP% = 72: GOTO 290
420  IF R1% = 2 THEN RT% = 48:HP%
      = 135:VP% = 144: GOTO 290
430  IF R1% = 3 THEN RT% = 32:HP%
      = 270:VP% = 72: GOTO 290
440 RT% = 16:HP% = 135:VP% = 9: GOTO
     290
450 LI% = LI% - 1
460  FOR L = 1 TO 3: POKE 768,200
     : POKE 769,100: CALL 770: NEXT
     L: POKE 768,250: POKE 769,25
     0: CALL 770
470  IF LI% >  - 1 THEN  GOTO 20
480  HTAB 1: VTAB 23: PRINT "ANOT
     HER GAME(Y/N)? ";: GET A$: IF
     A$ = "N" THEN  TEXT : HOME :
      VTAB 22: END
490  IF A$ <  > "Y" THEN  GOTO 48
     0
500 SC% = 0:LI% = 3:LE% = 1: GOTO
     20
510  VTAB 21: HTAB 8: PRINT SC%: VTAB
     21: HTAB 29: PRINT LE%: HTAB
     8: PRINT LI%: VTAB 22: HTAB
     29: CALL  - 868: VTAB 22: HTAB
     29: PRINT NG% - GE%: IF GE% =
     NG% THEN  POP : GOTO 360
520  RETURN
530  HGR :NS% = 0:SE% = 0:GE% = 0

540  HOME : VTAB 21: PRINT "SCORE
     :"; TAB( 21);"LEVEL :": PRINT
     "LIVES:"; TAB( 21);"GRAPES:"

550  HPLOT 130,3 TO 130,0 TO 0,0 TO
     0,67 TO 3,67 TO 3,3 TO 130,3
     : HPLOT 3,77 TO 0,77 TO 0,15
     3 TO 130,153 TO 130,150 TO 3
     ,150 TO 3,77
560  HPLOT 140,3 TO 140,0 TO 279,
     0 TO 279,67 TO 276,67 TO 276
     ,3 TO 140,3: HPLOT 140,150 TO
     140,153 TO 279,153 TO 279,77
      TO 276,77 TO 276,150 TO 140
     ,150
570 NG% = LE% * 3: IF NG% > 90 THEN
     NG% = 90
580  FOR L = 1 TO NG%
590 R1% = 9 * ( INT (30 *  RND (1
     ) + 1)):R2% = 9 * ( INT (16 *
      RND (1) + 1))
600  IF (R1% = 135 AND R2% = 72) OR
      (R1% = 9 AND R2% = 72) OR (R
     1% = 270 AND R2% = 72) OR (R
     1% = 135 AND R2% = 9) OR (R1
     % = 135 AND R2% = 144) THEN
      GOTO 590
610  FOR L2 = 1 TO L - 1
620  IF GH%(L2) = R1% AND GV%(L2)
      = R2% THEN 590
630  NEXT L2
640 GH%(L) = R1%:GV%(L) = R2%
650  IF  INT (10 *  RND (1) + 1) =
     5 THEN NS% = NS% + 1: DRAW S
     G% AT R1%,R2%: GOTO 670
660  DRAW GP% AT R1%,R2%
670  NEXT L
680  GOSUB 510:HP% = 135:VP% = 72
     :RT% = 0: DRAW GG% AT HP%,VP
     %
690 MH%(1) = 9:MV%(1) = 72:MH%(2)
      = 270:MV%(2) = 72:MH%(3) =
     135:MV%(3) = 9:NM% =  INT (3
      *  RND (1) + 1): FOR L = 1 TO
     NM%: DRAW MT% AT MH%(L),MV%(
     L): NEXT L
700  RETURN
710  TEXT : NORMAL : HOME
720  POKE 233,3: POKE 232,32
730  HCOLOR= 3: SCALE= 1: ROT= 0
740  DIM GH%(90),GV%(90)
750 IN% = 9:LI% = 3:LE% = 1:GG% =
     1:GP% = 2:SG% = 4:MT% = 3
760  FOR L = 0 TO 23: READ L2: POKE
     770 + L,L2: NEXT L
770  FOR L = 0 TO 191: READ L2: POKE
     800 + L,L2: NEXT L
780 A$ = " SOUR GRAPE": FOR L = 1
      TO 240: POKE 768,L: POKE 76
     9,7: CALL 770: PRINT A$;: NEXT
     L
790  FOR L = 10 TO 14: VTAB L: HTAB
     9: PRINT "
                ": NEXT L: VTAB 12
     : HTAB 9: PRINT "     BY CO
     MPUHOUSE      ": FOR L = 1 TO
     3000: NEXT L
800  HOME : RETURN
810  DATA  174,0,3,173,48,192,136
     ,208,5,206,1,3,240,9,202,208
     ,245,174,0,3,76,5,3,96
820  DATA  4,0,10,0,32,0,55,0,155
     ,0,146,10,63,28,28,28,36,197
     ,41,40,40,45,21,30,30,30,21,
     21,21,30,4,0,146,57,231,45,4
     5,12,63,63,63,44,45,45,37,63
     ,63,63,12,45,45,28
830  DATA  63,7,0,146,10,63,28,28
     ,28,36,197,41,40,40,173,21,2
     1,54,30,30,199,193,57,39,104
     ,6,0,0,12,0,146,57,231,28,36
     ,12,12,45,21,14,54,30,255,28
     ,45,45,28,31,199,41,45,229
840  DATA  31,7,0,255,255,12,12,4
     5,21,14,54,30,255,28,45,45,2
     8,31,199,41,45,229,31,7,0,25
     5,255,231,28,36,12,12,45,21,
     14,54,30,255,28,45,45,28,31,
     199,41,45,229,31,7,0,255
850  DATA  146,10,63,28,28,28,36,
     12,12,12,45,21,14,14,54,30,3
     0,31,199,105,197,249,31,199,
     105,13,13,28,31,31,12,13,28,
     7,0,255,255
```

考記性、考計數的益智教育遊戲

成雙成對

要教育兒童，最佳方法莫如從遊戲中入手。筆者設計的這個寓教育於娛樂的遊戲，相信會受到小學生的歡迎，做家長的不妨和孩子們一同玩。

這個「成雙成對」遊戲採用的形式是訓練兒童的記憶及意志集中力，再考驗他一些基本的加、減、乘及除數。

遊戲的玩法十分簡單。首先螢幕會出現兩欄格子，每欄各有十二小方格（見附圖），用來擺放12種不同的物件。物件是看不見的。玩者必須找出分別藏在兩欄內相同的一對物件，方法是按螢幕上顯示的橫列及縱列代號。例如A1及C6〔RET〕，按完〔RET〕後，被選中的方格子會現出放在那裏的物件。遊戲適宜兩人或兩隊以上進行大家輪流玩。假如選中一對的話，便有一個得分的機會，電腦會提出一條淺易的算術問題，玩者必須在限定時間之內算出結果，如此便可取得獎分，並且可以繼續玩下去。假如答錯，不但沒有獎分，還會倒扣五分，而機會會落回到對手那邊。

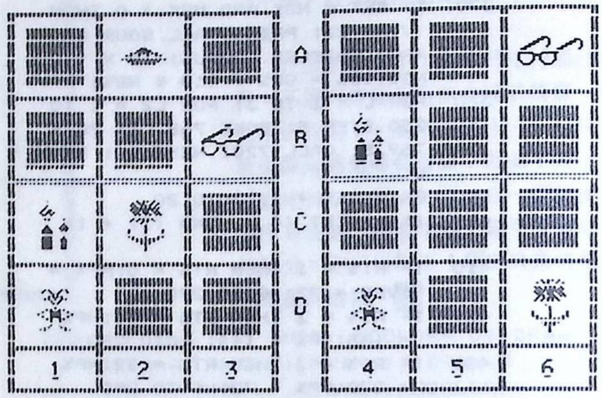
玩者在每個回合之中，有20秒的時間，這包括了選擇相同的一對物件，而計數那部份能有的時間長短則視乎這20秒內用來選擇方格子後所剩餘下來的時間。換言之，玩者必須要在20秒內完成整個回合。若然在法定時間內仍未能選對一對方格子，則會扣減15點積分，玩者必須緊記。

在選擇方格子時，有幾點必須不能犯錯：

(1) 輸入一個不在範圍內的代碼（例如E1或A9）
(2) 輸入的次序倒轉（例如3A、23或AD等）
(3) 選擇一個已開了的方格子。

假如作了上述任何一項不合規則的選擇，則會倒扣20分。

遊戲開始時，雙方都會有一百分基本分。當選擇權落回



對方手上時，中間會有10秒的過渡時間，只要按任何一個鍵，便可開始。當十二對物件完全被選對後，遊戲便結束，並會重新開始。不過，在下一個新回合裏，電腦出的算術問題會變得深些。基本上，這些算術是一些一個位、二個位及三個位的加、減、乘、除算術。

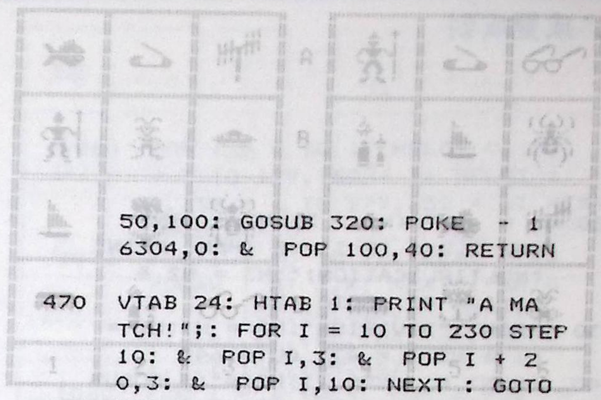
本遊戲共包括有兩個程式，其一是APPLESOFT，第二個是機械碼。APPLESOFT 程式部份大家可用下列指令來存放入磁碟：

SAVE MATCHING

至於機械碼部份是有關物件的圖形表。大家在輸入完後，可用下列指令存放入磁碟：

BSAVE M.SHAPES，A＄4000，L＄62B 至於機械碼的輸入方法，大家可參閱電腦時代第三期。■

```
10  REM   ***********************
20  REM   *      MATCHING        *
30  REM   *    BY  COMPUHOUSE    *
40  REM   * COPYRIGHT (C) 1985 *
50  REM   * BY COMPUTING AGE     *
60  REM   *                      *
70  REM   ***********************
80  REM
90  LOMEM: 24576: GOTO 760
100 REM
110 II = II - 1:TT =  INT (II / 1
    5): IF II / 15 <  > TT THEN
     RETURN
120  &  POP 75,2:J = TT > 9: IF T
    T = 5 THEN FF = 112
130  POKE 1892,TT - 10 * J + FF: IF
    TT < 20 THEN  POKE 1891,FF +
    J
140  IF TT THEN  RETURN
150 OF = 1: POP : RETURN
160 I =  FRE (0):HP =  POS (0) +
    1:I$ = ""
170  INVERSE : PRINT " ";: NORMAL
    : PRINT BS$;
180 I =  PEEK (KB) - 128: IF TF THEN
     GOSUB 110
190  IF I < 0 THEN 180
200  POKE KR,0: &  POP I + 50,2:A
    $ =  CHR$ (I): IF I = 13 AND
     LEN (I$) THEN  PRINT " ";:I
     =  FRE (0): RETURN
210  IF ( LEN (I$) > 6 OR I < 33)
     AND I <  > 8 THEN 170
220  IF I <  > 8 THEN  PRINT A$;:
     I$ = I$ + A$: GOTO 170
230  IF TF THEN  GOSUB 110: GOSUB
     110
240  IF  LEN (I$) < 2 THEN  HTAB
     HP: PRINT "  ";: HTAB HP: GOTO
     160
250  PRINT BS$"  "BS$BS$;:I$ =  LEFT$
     (I$, LEN (I$) - 1): GOTO 170
260 A =  ASC ( LEFT$ (I$,1)) - 64
    :B =  VAL ( RIGHT$ (I$,1))
270  IF A < 1 OR A > 4 OR B < 1 OR
     B > 6 OR  LEN (I$) <  > 2 THEN
     EF = 1: RETURN
```




```
280 C = L(A,B): IF M1 = A AND M2 =
     B OR C =  - 1 THEN EF = 1: RETURN

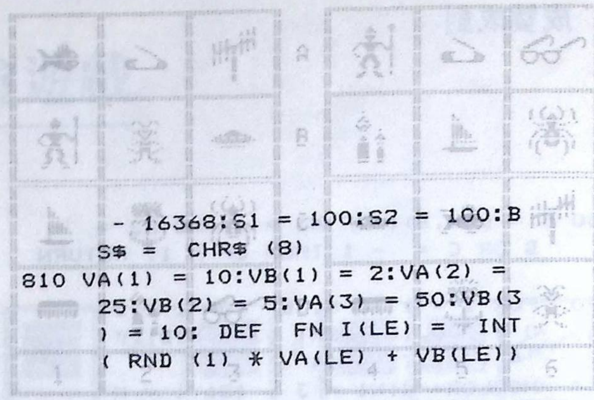
290  FOR I = 1 TO 5: &  POP I * 1
     0 + 200,10: NEXT :J = X(B):K
      = Y(A):T1 = 14
300  HCOLOR= 4:X1 = J - 18:Y1 = K
      - 14: FOR I = Y1 + 15 TO Y1
      + 30: HPLOT X1,I TO X1 + 36
     ,I
310  HPLOT X1,I + 14 - T1 TO X1 +
     36,I + 14 - T1:T1 = T1 + 2: NEXT
     : HCOLOR= SC(C): DRAW C AT J
      - DX(C),K - DY(C): RETURN
320  FOR J = 1 TO 250: NEXT : RETURN

330  FOR J = 1 TO 500: NEXT : RETURN

340 EF = 0:FF = 176:II = 299:M1 =
     0:M2 = 0:OF = 0: VTAB 23: HTAB
     20: PRINT "20"
350  FOR I = 21 TO 24: VTAB I: HTAB
     1: PRINT  SPC( 18): NEXT : VTAB
     21: HTAB 1: RETURN
360 A = M1:B = M2: GOSUB 410:A =
     M3:B = M4: GOSUB 410
370  CALL 54915: ON PN GOTO 750,7
     40
380  VTAB 23: IF PN = 1 THEN S1 =
     S1 + TS:S1 = S1 * (S1 > 0):I
      = S1
390  IF PN = 2 THEN  PRINT :S2 =
     S2 + TS:S2 = S2 * (S2 > 0):I
      = S2
400  HTAB 36: PRINT  RIGHT$ ("000
     " +  STR$ (I),4);: RETURN
410  &  POP 150,10: &  POP 200,10
     : &  POP 250,10:J = X(B):K =
     Y(A):X1 = J - 18:X2 = J - 14
     :Y1 = K - 14:T1 = 30:T2 = 22

420  FOR I = Y1 TO Y1 + 15: HCOLOR=
     4: HPLOT X1,I TO X1 + 36,I: HPLOT
     X1,I + T1 TO X1 + 36,I + T1:
     T1 = T1 - 2: IF I < K - 10 THEN
      NEXT
430  HCOLOR= 5: HPLOT X2,I TO X2 +
     28,I: HPLOT X2,I + T2 TO X2 +
     28,I + T2:T2 = T2 - 2: NEXT
     : RETURN
440  POKE KR,0: VTAB 12: HTAB 20:
      PRINT "09": POKE  - 16303,0
     : GOSUB 330:I = 100
450  IF  PEEK (KB) < 128 THEN I =
     I - 1: IF I THEN  &  POP 150
      - I,3: POKE 1468, INT (I /
     10) + 176: GOTO 450
460  POKE KR,0: GOSUB 320: &  POP
     50,100: GOSUB 320: POKE  - 1
     6304,0: &  POP 100,40: RETURN

470  VTAB 24: HTAB 1: PRINT "A MA
     TCH!";: FOR I = 10 TO 230 STEP
      10: &  POP I,3: &  POP I + 2
     0,3: &  POP I,10: NEXT : GOTO
     330
480  VTAB 24: HTAB 1: PRINT "CORR
     ECT!";: FOR I = 10 TO 150: &
      POP I,2: NEXT : GOSUB 330:T
     S = 25 + TT * 5: GOTO 380
490  VTAB 24: HTAB 1: PRINT "WRON
     G ANSWER!";: FOR I = 10 TO 1
     50 STEP 10: &  POP I,10: &  POP
     1,15: NEXT : GOSUB 330:TS =
      - 5: GOTO 380
500  VTAB 24: HTAB 1: PRINT "SORR
     Y, NO MATCHES!";: FOR I = 1 TO
     5: &  POP 100,20 + I: &  POP
     150,20 - I: NEXT : GOSUB 330
     :TS =  - 10: GOTO 380
510  VTAB 24: HTAB 1: PRINT "TIME
     'S UP";: FOR I = 200 TO 100 STEP
      - 10: &  POP I,10: &  POP 5
     0,5: GOSUB 330:TS =  - 15: GOTO
     380
520  VTAB 24: HTAB 1: PRINT "ILLE
     GAL MOVE!";: FOR I = 1 TO 5:
      &  POP 170 + 10 * I,60 + I:
     NEXT : GOSUB 330:TS =  - 20
     : GOTO 380
530  VTAB 21: HTAB 1: GOSUB 350: IF
     S1 = S2 THEN  FLASH : PRINT
     "  TIE GAME  ": NORMAL : GOTO
     560
540  IF S1 > S2 THEN  INVERSE : PRINT
     P1$;: NORMAL : PRINT " DEFEA
     TS": PRINT P2$", "S1" TO "S2
     : GOTO 560
550  PRINT P2$" DEFEATS": INVERSE
     : PRINT P1$;: NORMAL : PRINT
     ", "S2" TO "S1
560  VTAB 24: PRINT "PLAY AGAIN?
     ";: GET A$: PRINT A$: IF A$ <
      > "N" THEN  RUN
570  TEXT : HOME : END
580  GOSUB 350:N1 =  FN I(LE):N2 =
      FN I(LE):I =  INT ( RND (1)
      * 4) + 1: ON I GOTO 590,600
     ,620,620
590  PRINT "ADDITION:":N3 = N1 +
     N2:I = 43: GOTO 640
600  PRINT "SUBTRACTION:": IF N2 >
     N1 THEN I = N2:N2 = N1:N1 =
     I
610 N3 = N1 - N2:I = 45: GOTO 640
```




```
620 N1 =  INT (N1 / 2):N2 =  INT
    (N2 / 2): IF I = 3 THEN  PRINT
    "MULTIPLICATION:":N3 = N2 *
    N1:I = 42: GOTO 640
630 PRINT "DIVISION":N3 = N1:N1 =
    N1 * N2:I = 47: GOTO 640
640 PRINT N1 CHR$ (I)N2"=";: GOSUB
    160: IF OF THEN  GOSUB 510: GOTO
    360
650 IF  VAL (I$) <  > N3 THEN  VTAB
    23: HTAB HP: FLASH : PRINT N
    3: NORMAL : GOSUB 490: GOTO
    360
660 GOSUB 480:L(M1,M2) =  - 1:L(
    M3,M4) =  - 1:CS = 0: RETURN
670 PRINT "'S TURN": IF CS THEN
     GOSUB 440
680 CS = 1: VTAB 22: HTAB 1: PRINT
    "MOVE 1: ";: GOSUB 160: IF O
    F THEN  GOSUB 510: GOTO 370
690 GOSUB 260:M1 = A:M2 = B: IF
    EF THEN  GOSUB 520: GOTO 370
700 VTAB 23: HTAB 1: PRINT "MOVE
    2: ";: GOSUB 160: IF OF THEN
     GOSUB 510:A = M1:B = M2: GOSUB
    410: GOTO 370
710 GOSUB 260:M3 = A:M4 = B: IF
    EF THEN  GOSUB 520:A = M1:B =
    M2: GOSUB 410: GOTO 370
720 IF L(M1,M2) <  > L(M3,M4) THEN
     GOSUB 500:A = M1:B = M2: GOSUB
    410:A = M3:B = M4: GOSUB 410
    : GOTO 370
730 GOSUB 470: GOTO 580
740 GOSUB 340:PN = 1: INVERSE : PRINT
    P1$;: NORMAL : GOSUB 670:NU =
    NU + 1: ON (NU = 12) + 1 GOTO
    740,530
750 GOSUB 340:PN = 2: PRINT P2$;
    : GOSUB 670:NU = NU + 1: ON
    (NU = 12) + 1 GOTO 750,530
760 TEXT : HOME : IF  PEEK (1638
    4) <  > 32 THEN  PRINT  CHR$
    (4)"BLOAD M.SHAPES"
770 POKE 48,170: HLIN 9,30 AT 8:
     HLIN 9,30 AT 20: VLIN 9,19 AT
    8: VLIN 9,19 AT 31
780 VTAB 7: HTAB 12: PRINT "
     MATCHING       ": PRINT : HTAB
    14: PRINT "BY COMPUHOUSE "
790 POKE 1013,76: POKE 1014,0: POKE
    1015,64: POKE 230,32: POKE 2
    32,48: POKE 233,64: CALL 624
    50
800 I = 12: DIM A(I),B(I),DX(I),D
    Y(I),SC(I):KB =  - 16384:KR =
     - 16368:S1 = 100:S2 = 100:B
    S$ =  CHR$ (8)
810 VA(1) = 10:VB(1) = 2:VA(2) =
    25:VB(2) = 5:VA(3) = 50:VB(3
    ) = 10: DEF  FN I(LE) =  INT
    ( RND (1) * VA(LE) + VB(LE))
820 FOR I = 1 TO 3:X(I) = 22 + (
    I - 1) * 42:X(I + 3) = 150 +
    X(I): NEXT
830 FOR I = 1 TO 4:Y(I) = 19 + (
    I - 1) * 33: NEXT : FOR I =
    1 TO 12: READ SC(I),DX(I),DY
    (I): NEXT
840 DATA 3,8,6,3,9,9,3,9,3,7,7,8
    ,3,9,2,7,9,4,3,7,8,3,14,4,3,
    9,9,7,8,6,3,10,9,7,10,2
850 FOR I = 1 TO 12:A(I) = I:B(I
    ) = I: NEXT : FOR I = 1 TO 1
    2:J =  INT ( RND (1) * 12 +
    1)
860 K =  INT ( RND (1) * 12 + 1):
    T = A(I):A(I) = A(J):A(J) =
    T:T = B(I):B(I) = B(K):B(K) =
    T: NEXT
870 FOR J = 1 TO 3: FOR K = 1 TO
    4:I = K + (J - 1) * 4:L(K,J)
     = A(I):L(K,J + 3) = B(I): NEXT
    : NEXT : POKE KR,0
880 VTAB 22: HTAB 4: PRINT "->
     PRESS ANY KEY TO BEGIN
    <-";
890 FOR I = 1 TO 3: &  POP 100 -
    I,2: HTAB 3 + I: PRINT "->";
    : IF  PEEK (KB) > 128 THEN 9
    30
900 HTAB 37 - I: PRINT "<-";: GOSUB
    320: NEXT : HTAB 4: PRINT  SPC(
    4): HTAB 34: PRINT  SPC( 4)
910 FOR I = 1 TO 3: &  POP 100 -
    I,2: HTAB 7 - I: PRINT "<-";
    : IF  PEEK (KB) > 128 THEN  POKE
    KR,0: CALL 54915: GOTO 930
920 HTAB 33 + I: PRINT "->";: GOSUB
    320:J =  RND (1): NEXT : GOTO
    880
930 HOME : INVERSE : PRINT  SPC(
    11)"MATCHING.....   " SPC(
    12)"COPYRIGHT (C) 1985  BY C
    OMPUTING AGE    ": NORMAL
940 VTAB 5: PRINT "      MATCHI
    NG     IS THE TIMELESS GAME O
    F CONCENTRATION WITH AN ADDE
    D TWIST."
950 INVERSE : HTAB 6: PRINT "PLE
    ASE REFER TO THE MAGAZINE": NORMAL
980 PRINT : PRINT "ENTER YOUR NA
    MES AND WE'LL BEGIN...": VTAB
```

成雙成對

```
     19: HTAB 1: PRINT "PLAYER 1
     'S NAME? ";: GOSUB 160:P1$ =
     I$
990  VTAB 21: HTAB 1: CALL  - 868
     : PRINT "PLAYER  2'S NAME? "
     ;: GOSUB 160:P2$ = I$: IF P1
     $ = P2$ THEN 990
1000  VTAB 23: HTAB 1: INVERSE : PRINT
     "1=EASY 2=MODERATE 3=HARD": NORMAL
1010  VTAB 24: HTAB 1: CALL  - 86
     8: PRINT "WHAT LEVEL (1-3)?
     ";: GOSUB 160:LE =  VAL (I$)
     : IF LE < 1 OR LE > 3 THEN 1
     010
1020  GOSUB 330: &  POP 81,32: &
     POP 77,128: &  POP 81,128: &
     POP 102,64: &  POP 121,128:
     &   POP 91,64: GOSUB 320
1030 TF = 1: HOME : HGR : SCALE=
     1: ROT= 0: HCOLOR= 6
1040  FOR I = 3 TO 135 STEP 33: FOR
     J = 0 TO 1: HPLOT 0,I + J TO
     129,I + J: HPLOT 150,I + J TO
     279,I + J: NEXT : NEXT
1050  FOR I = 0 TO 126 STEP 42: FOR
     J = 0 TO 2: HPLOT I + J,3 TO
     I + J,153: HPLOT I + J + 150
     ,3 TO I + J + 150,153: NEXT
     : NEXT
1060  FOR I = 0 TO 1: HPLOT 0,152
     + I TO 129,152 + I: HPLOT 1
     50,152 + I TO 279,152 + I: NEXT
     : HCOLOR= 5
1070  FOR I = 13 TO 22: READ J,K:
     XDRAW I AT J,K: NEXT : DATA
     137,17,137,50,137,83,137,116
     ,21,141,62,141,104,141,170,1
     41,212,141,254,141
1080  FOR I = 1 TO 4: FOR J = 1 TO
     6:X1 = X(J) - 14:Y1 = Y(I) -
     10:L = 22: &  POP 200,5
1090  FOR K = Y1 TO Y1 + 11: HPLOT
     X1,K TO X1 + 28,K: HPLOT X1,
     K + L TO X1 + 28,K + L:L = L
      - 2: NEXT : NEXT : NEXT
1100  VTAB 21: HTAB 19: PRINT "TI
     ME";: HTAB 26: PRINT "PLAYER
     'S SCORES": HTAB 20: PRINT "
     20";: HTAB 26: INVERSE
     PRINT P1$;: NORMAL : HTAB 3
     4: PRINT "-";: HTAB 36: PRINT
     "0100": HTAB 26: PRINT P2$;:
     HTAB 34: PRINT "-";
1120  HTAB 36: PRINT "0100";: VTAB
     8: HTAB 7: PRINT "PRESS ANY
     KEY TO RESUME PLAY": POKE 48
     ,173: HLIN 6,33 AT 16: POKE
     34,20: GOTO 740
```

學校電腦學會活動走廊通訊徵稿

各位同學，我們計劃開設一個〔學校電腦學會活動走廊〕，旨在讓不同學校的同學能透過本走廊互相交流彼此學習電腦所遭遇的經驗、趣事，或報導自己學校電腦學會的消息活動。我們希望同學們能踴躍投稿。

來稿以每則不超過二百字爲合，內容不拘，可以是報導電腦會活動消息，上電腦課時所遭遇的趣事、或電腦學會所發生的趣聞等等。一旦採用，每則送出高級磁碟一張。

來稿請用原稿紙打橫書寫，註明所就讀的學校班級，附上眞實姓名、聯絡電話及地址寄來九龍中央郵政信箱CPO BOX 71193 號。發表時可用筆名。來稿信封請勿封口，並請註明〔學校電腦活動走廊」投稿。

電腦學會活動贊助基金申請

我們計劃設立一項〔學校電腦學會活動贊助基金〕，歡迎各校電腦學會參加申請。這項贊助基金的設立目的，旨在提供擴活動及增添設備經費之用，俾學會同學能加強對電腦知識的認識及能藉此令各校電腦學會之間互相交流經驗。

獲取獎金方法

電腦時代將設立一欄〔學校電腦學會活動通訊走廊〕，各電腦學會可申請成爲基本通訊員。凡屬於基本通訊員的同學投寄來活動走廊欄的稿件一經刊出，即可取得分點，每則一分，累積六期所登，獲最高分點的通訊員將可獲取五百元港幣的活動贊助獎金，如有通訊員取得相同分數，獎金則均分。

申請成爲基本通訊員辦法簡單，學會同學可來函九龍中央郵政信箱71193號索取申請表格及有關章程。

所有申請者必須以學校電腦學會名義，否則不獲接納。■

成雙成對

```
]

****************
*   M.SHAPE    *
* A$4000,L$62B*
****************

4000- 20 B1 00 20 B9 F6 86 06
4008- 85 09 85 07 C8 84 08 2C
4010- 30 C0 88 D0 04 C6 07 F0
4018- 07 CA D0 F6 A6 06 D0 EF
4020- A5 09 85 07 C6 08 D0 F1
4028- 60 00 00 00 00 00 00 00
4030- 16 00 2E 00 9A 00 30 01
4038- 7B 01 E6 01 25 02 81 02
4040- 07 03 8E 03 2C 04 8A 04
4048- 2D 05 6A 05 7A 05 89 05
4050- 97 05 A6 05 B0 05 BF 05
4058- CE 05 DC 05 EC 05 49 0D
4060- 4D 49 0D 4D DA DF FB 1B
4068- DF FB 4A 49 09 0D 4D 49
4070- D1 DB FB 1F 1F DF 9B 49
4078- 0D 4D 4D 0D 4D DA DB FB
4080- 1B DF DB 4A 49 09 0D 4D
4088- 49 D1 DB DB FB DB DB 4A
4090- 09 0D 4D 69 4D D1 FB 1B
4098- 3F 3F DF FB 6A 49 29 2D
40A0- 6D 49 F1 FB 1B 3F 3F DF
40A8- FB 4A 09 0D 4D 69 4D D1
40B0- DB 1B DF FB DB 53 49 09
40B8- 4D 69 49 D1 1B 1F DF DB
40C0- 1F DF 0A 0D 4D 49 49 69
40C8- 15 00 49 49 49 49 49 49
40D0- 31 DF DF DB DB DB DB 53
40D8- 49 49 49 49 4D 4D FE FB
40E0- FB DB DB DB 9B 49 49 4D
40E8- 49 69 69 35 1F 3F 1F DF
40F0- DB FB FB 13 4D 4D 4D 49
40F8- 2D 0D 4D FE FB FB 3F FF
4100- DF DF 6E 69 2D 0D 6D 69
4108- 69 F1 1B DF DF FF FB 3F
4110- 1F 2E 0D 4D 4D 6D 69 49
4118- D1 DB DB 1B FF FB FB 33
4120- 4D 4D 4D 6D 49 49 D1 DB
4128- DB 1B FF DB FB 33 4D 49
4130- 49 6D 49 49 D1 DB DB 1B
4138- FF DB DB 53 49 49 09 6D
4140- 49 49 D1 DB DB 1B FF DB
4148- DB 53 49 49 09 6D 49 49
4150- D1 DB DB 1B FF DB DB 53
4158- 49 49 09 6D 49 49 11 00
4160- 49 09 6D 49 49 49 89 DB
4168- DB DB 1B FF DB 9B 49 49
4170- 49 6D 49 49 D1 DB DB FF
4178- DB DB DB 4A 49 49 49 09
4180- 6D 49 DA 3B DF DB DB 3B
4188- 3F BF 2D 6D 49 49 49 09
4190- 2D 15 FF FF DB DB DB DF
4198- 37 2D 2D 4D 49 49 09 6D
41A0- F5 3F 3F 3F 3F 3F 3F 3F
41A8- 3F BF 00 49 4D 49 49 D1
41B0- DB DB DB DB 4A 69 49 49
41B8- 89 DB DB DB 1B 9F 69 29
41C0- 4D 49 89 DB DB FB FF 73
41C8- 6D 69 49 49 DA DF DB FB
41D0- 9F 49 49 49 49 D1 FB DB
41D8- DB DB 4A 69 49 09 0D 15
41E0- 3F 1F DF 1B 3F 9F 29 2D
41E8- 6D 49 0D D5 DB DB 3B 3F
41F0- BF 29 2D 6D 49 69 1A 3F
41F8- DF 1B 3F 3F 57 2D 2D 4D
4200- 09 2D D5 3F DF 1B 3F 3F
4208- 57 2D 2D 4D 09 2D D5 3F
4210- DF 1B 3F 3F 17 00 29 2D
4218- 2D 2D 2D 2D 2D 2D AD 3F
4220- 3F 3F 3F 3F 3F 3F 3F 3F
4228- 2E 0D 0D 0D 0D 0D 0D 0D
4230- 0D 35 FF 1F 1F 1F 1F 1F
4238- 1F 1F 37 6D 0D 0D 0D 0D
4240- 0D 0D 0D 35 FF 1F 1F 1F
4248- 1F 1F 1F 1F 77 4D 69 09
4250- 4D 69 09 15 00 49 49 49
4258- 49 0D 4D 89 DB 1B 1F 1F
4260- DF DB DB 13 2D 4D 49 09
4268- 0D 2D 6D 89 1B 3F 3F 3F
4270- FF DF 1B 3F 57 29 6D 09
4278- 2D 2D 2D 6D A9 3F 3F FF
4280- 3F 3F 3F 3F FF 53 09 2D
4288- 2D 2D 2D 6D 2D 2D 15 DF
4290- 3B 1F 3F 3F 3F 3F FF 53
4298- 29 6D 09 2D 2D 6D 2D AD
42A0- 1B FF 3F 3F DF DB 3B BF
42A8- 2D 4D 49 49 29 2D 4D 11
42B0- 00 49 69 69 4D 4D D1 FB
42B8- FB 1B 1F DF 9F 69 4D 0D
42C0- 0D 4D 4D DA 1F 1F 1F 1F
42C8- 1F 1F 9F 69 0D 0D 2D 0D
42D0- 0D 4D DA 1F DF 3F DF 1F
42D8- 9F 49 0D 4D 0D 4D 0D D5
42E0- DB FB 1F 1F DF 9F 69 69
42E8- 69 4D 4D D1 DB DB DB DF
42F0- DB 4A 49 49 4D 49 89 DB
42F8- DB 1B DF DB 9B 4D 49 09
4300- 4D 49 09 FE DB DB DF DB
4308- 1B 6E 4D 49 4D 49 0D DE
4310- 1F 1F 1F 1F 1F 1F 9F 49
4318- 69 0D 0D 0D 4D D1 DB 1B
4320- 1F 1F DF DB 4A 49 49 4D
4328- 49 89 DB DB 1B DF DB 9B
4330- 49 49 09 4D 49 89 00 49
4338- 49 49 09 2D 4D 49 49 49
4340- 49 D1 3F DF DB DB DB DF
4348- 3F DF DB DB 4A 49 09 2D
4350- 4D 09 4D 49 49 29 6D 31
4358- DF 1B 3F DF DB 1B DF DB
4360- 3B FF DB 4A 29 6D 49 49
4368- 4D 49 29 6D 49 09 1E DF
4370- DB 1B 3F DF DB DB DB DB
4378- 3F 17 2D 2D 2D 4D 09 2D
4380- 2D 2D 6D 49 49 49 DA DB
4388- DB DB FB DB 1B 3F FF DB
4390- 1B 6E 49 09 4D 4D 49 69
4398- 49 49 49 D1 DB DB DB 1B
43A0- DF DB FB FB DB 1B 0E 4D
43A8- 09 4D 09 4D 09 4D 49 49
43B0- 49 D1 DB DB DB DB 3B 3F
43B8- DF DB 3B 3F 9F 00 49 49
43C0- 49 49 49 49 15 3F DF DB
43C8- DB DF DB 9B 49 49 29 6D
43D0- 49 49 A9 FB DB DB 1F 1F
43D8- DF DB 4A 49 29 2D 6D 49
43E0- 49 D5 DF DB DB 1F DF DB
43E8- 53 49 49 69 49 49 09 D5
43F0- DF 1B 3F 3F 3F 3F 3F DF
43F8- 4A 4D 2D 2D 2D 4D 4D A9
4400- 3B 1F DF 1B 3F 3F DF 1B
4408- 17 4D 49 29 6D 49 09 0D
4410- D5 DF DB DB 1F DF DB 57
4418- 29 4D 2D 2D 4D 49 09 D5
4420- DF DB 3B FF 3F DF 9B 49
4428- 09 6D 09 6D 49 09 D5 DF
4430- DB FF DB 3B DF 53 49 6D
4438- 49 29 4D 49 D5 DF DB 3B
4440- DF 3B DF 9B 49 09 6D 09
4448- 6D 49 09 D5 DF 1B 3F DF
4450- DB 3F DF 4A 2D 4D 49 09
4458- 2D 4D 89 00 49 69 49 49
4460- 49 DA DB DB DB DF 53 49
4468- 0D 4D 49 49 DA DB DB FB
4470- DF 53 49 0D 0D 4D 49 D1
4478- DB DB 1F 1F DF 53 49 0D
4480- 0D 0D 4D 89 DB 1B 1F 1F
4488- 1F DF 53 49 0D 0D 0D 0D
4490- 4D DA FB 1F 1F 1F DF 53
4498- 49 0D 0D 0D 0D 0D D5 DB
44A0- DB DB FB 9B 49 69 49 49
44A8- 49 3A 3F 3F 3F 3F 3F 3F
44B0- 3F 4E 29 2D 2D 2D 2D 6D
44B8- 02 00 49 49 69 49 69 49
44C0- 89 FB DB DF DB 1B DF 1B
44C8- 17 4D 09 4D 49 49 69 49
44D0- FE DB DF 1B DF 1B DF 1B
44D8- 6E 49 4D 29 6D 09 4D 09
44E0- FE DB FB 3F 3F 1F DF DB
44E8- 4E 4D 49 2D 2D 4D 49 8D
44F0- DB FB DF 3F DF 1F DF 53
44F8- 49 49 09 2D 4D 49 49 DA
4500- DB 1F 1F 3F 1F 1F DF 9B
4508- 49 4D 09 6D 6D 49 4D DA
4510- DF 1B FF 1B FF DB 9F 69
4518- 09 0D 2D 0D 2D 0D 4D A9
4520- DF FB 3B DF 3B DF DF 33
4528- 4D 4D 29 6D 2D 4D 69 31
4530- DF DF 1B 3F 3F DF 1B DF
4538- 6E 69 49 49 49 49 4D DE
4540- FB DB DB DB DB DF 4A 69
4548- 49 49 49 49 4D DA 1B DF
4550- DB DB DB DF 53 49 69 49
4558- 49 09 4D 89 00 49 49 49
4560- 2D 4D 49 49 DA DB 1B 3F
4568- 3F 3F DF DB 53 49 0D 0D
4570- 0D 0D 0D 0D 4D D1 FB DF
4578- FB 1B DF FB 9F 0D 0D 0D
4580- 0D 0D 0D 0D 0D 0D 0D DE
4588- FB 1F 1F 1F 1F 1F 1F DF
4590- 4A 49 0D 0D 0D 0D 0D 4D
4598- 89 00 09 8D FB 17 4D 31
45A0- DF 33 2D 2D FE 1B 6E 09
45A8- 06 00 2D AD DF 33 4D F1
45B0- 3F 37 4D 31 DF 33 2D AD
45B8- 00 29 AD DF 33 4D D1 DB
45C0- 6E 89 DF 73 2D 15 00 2D
45C8- AD DF 33 4D 31 DF 33 4D
45D0- 31 DF 33 2D AD 00 A9 3B
45D8- 0E D5 57 D5 17 2D 06 00
45E0- 29 AD DF 73 49 1E FF 0A
45E8- 4D DA 1B 2E 2D 35 00 2D
45F0- 2D FE 9B 49 D5 FF 4A 09
45F8- FE 1B 0E 2D 15 00 49 D5
4600- FF 0A 0D D5 DF 2E 2D F5
4608- DF 4A A9 00 2D 2D DE 1B
4610- 2E 2D 15 DF 53 49 FE 1B
4618- 0E 2D 15 00 09 2D DE BB
4620- 4D D1 3F 37 4D 31 DF 73
4628- 2D 15 00 00
```

■

能計算七十個未知數

組合線性方程式解答系統

各位同學，你們有沒有試過要計算一大堆的線性方程式，尋求各個未知數的答案。假如只有幾個未知數，計算時還容易，大家可用矩陣方法來幫助快速計算。不過，假如未知數超過10個或甚至有五十個，則如仍用筆算，花數星期也未必能將正確的結果全部計算出。

筆者編寫了一個專用來運算線性方程式的程式來幫助大家解決。程式用 APPLESOFT 寫成，可以運算最多達七十條線性方程式（最多七十個未知數）。

現在先讓筆者講述本程式的用法。假設有三條線性方程式：

(1) $5X+3Y=22$

(2) $X-Y+Z=-1$

(3) $X+Z=5$

在利用本程式時，首先要決定是否要將結果輸出往印字機。然後電腦會問有多少條方程式，在本例子中是3條。接著電腦會問總共有多少個變數要計算，在本例子亦是有3個（即X，Y及Z）。

接著開始進行輸入工作。在此，必須留意一項事，就是大家必須先將每一條方程式內的未知數，順次序排列好。假如方程式內沒有該變數，則必須要補上0，在本例子中，應視作如下：

(1) $5X+3Z+0Z=22$

(2) $1X-1Y+1Z=-1$

(3) $1X+0Y+1Z=5$

大家只要輸入系數及常數（一定要順序）即可。電腦會首先要求你輸入第一條方程式的系數。在本例子中，第一條方程式的系數及常數是5，3，0及22。先按5〔RET〕，3〔RET〕，0〔RET〕，及22〔RET〕。然後是第二條方程式，1，－1，1，－1。最後是第三條方程式：1，0，1，5。

當所有系數輸入後，電腦便會將這三條方程式列出。當然，電腦不會辨認三條式中的未知數是X，Y或Z，因爲筆者爲著令同學方便輸入起見，假定了所有的未知數爲X1，X2，X3，X4，……XN。

此時電腦所列出的例子程式：

(1) $5*X1+3*X2+0*X3=22$

(2) $1*X1+-1*X2+1*X3=-1$

(3) $1*X1+0*X2+1*Z=5$

然後，螢幕會詢問所有輸入是否正確，這裏提供你更正的機會。更正方式十分容易，大家只要照螢幕的指示來做便可。這裏不再詳述。

當一切沒有問題，便可計算出答案。利用本程式，就算有七十條方程式包括七十個未知數，也不用再怕，閣下可以在極短的時間內計算出答案。

最後，筆者要再次強調，本程式只適用於解決線性方程式的未知數。不過，有某些情況下，仍可計算一些二次元方程式，這裏筆者留待各位同學自行去思考了！

打入這程式相當簡單，大家只要按列表打入即可，然後用下列指令將之存入磁碟上：

SAVE EQUATION SOLVER

```
10  REM  ***********************
20  REM  ** EQUATION SOLVER **
30  REM  ** COPYRIGHT 1985  **
40  REM  **      BY         **
50  REM  **  COMPUTING AGE  **
60  REM  ***********************
70  REM
80  HOME : VTAB 9: INVERSE : POKE 32,9: PRINT : PRINT "   EQUATION SOLVER   "
100  POKE 32,0: PRINT : NORMAL
110  VTAB 20: HTAB 10: PRINT "PRESS ANY KEY TO BEGIN": VTAB 22: PRINT "** COPYRIGHT 1985 BY COMPUTING AGE   **": WAIT  - 16384,128: POKE  - 16368,0
130  REM  GLOSSARY OF VARIABLES
150  REM  M(M,N+1) WILL BE AN ARGUMENT MATRIX FOR A SYSTEM OF LINEAR EQUATIONS
160  REM  K$ IS A QUESTION RESPONSE
170  REM  B,I,J,K,R,S,T,U, AND V ARE COUNTERS
180  REM  L IS A CONSTANT
190  REM  TEMP(N+1) WILL BE A TEMPORARY STORAGE ARRAY FOR SWITCHING EQUATIONS
200  REM  M IS THE # OF ROWS OR EQUATIONS
210  REM  N IS THE # OF VARIABLES IN THE EQUATIONS (N+1= # OF COLUMNS)
220  REM  FLAG IS USED TO CHECK CONDITIONS
230  REM  A IS THE # OF ZERO ROWS
240  REM
250  DIM M(71,72): DIM TEM(72)
270  REM  MAIN PROGRAM
290  HOME
300  PRINT "WOULD YOU LIKE THE RESULTS PRINTED        (Y OR N)? ";: GET P$
310  IF P$ <  > "Y" THEN 360
320  PRINT : PRINT
330  PRINT "WHAT SLOT# IS THE PRINTER IN?";: INPUT " ";SLN$
340 SLN =  INT ( VAL (SLN$))
350  IF SLN > 6 OR SLN < 1 THEN 330
```

組合線性方程式解答系統

```
360  HOME
370  PRINT "EQUATIONS MUST BE IN
     THE FORM           A1*X1 + A
     2*X2 +...+ AN*XN =B
380  PRINT
390  PRINT "HOW MANY EQUATIONS AR
     E THERE?"
400  INPUT "M= ";M
410  PRINT : PRINT "HOW MANY VARI
     ABLES ARE THERE?"
420  INPUT "N= ";N
430 M =  INT (M):N =  INT (N)
440  IF M > 70 OR N > 70 THEN  HOME
     : PRINT "THE LARGEST POSSIBL
     E VALUE FOR M AND N  IS 70.
      TRY AGAIN.": FOR K = 1 TO 4
     000: NEXT K: HOME : GOTO 390
450  IF M <  = 0 OR N <  = 0 THEN
      HOME : PRINT "M AND N MUST
     BOTH BE LARGER THE ZERO.    T
     RY AGAIN.": FOR K = 1 TO 400
     0: NEXT K: HOME : GOTO 390
460  PRINT
470  PRINT "INPUT THE COEFFICIENT
     S AND CONSTANTS BY  ROW (EQU
     ATIONS)."
480  FOR I = 1 TO M
490  PRINT
500  PRINT "INPUT THE COEFFICIENT
     S AND CONSTANTS FOREQUATION
     #";I;"."
510  FOR J = 1 TO N + 1
520  INPUT M(I,J)
530  NEXT J
540  NEXT I
550 D$ =  CHR$ (4)
560 K$ = "N"
570  IF P$ = "Y" AND K$ = "Y" THEN
      PRINT : PRINT D$;"PR#"SLN: GOTO
     590
580  HOME
590  PRINT "THE SYSTEM OF EQUATIO
     NS IS:": PRINT
600  FOR U = 1 TO M
610  IF P$ = "Y" AND K$ = "Y" THEN
     640
620 CV =  PEEK (37): REM PEEK(37)
      IS THE VALUE OF THE CURSOR'
     S CURRENT VERTICAL POSITION
630  IF CV > 20 THEN  GOSUB 2120
640  PRINT "EQU. #";U
650  PRINT
660  FOR V = 1 TO N + 1
670  IF P$ = "Y" AND K$ = "Y" THEN
     700
680 CV =  PEEK (37)
690  IF CV > 20 THEN  GOSUB 2120
700  IF V = N + 1 THEN  PRINT " =
     ";M(U,N + 1)
710  IF V = 1 THEN  PRINT M(U,V);
     "*X1";
720  IF V > 1 AND V <  > N + 1 THEN
      PRINT "+";M(U,V);"*X";V;
730  NEXT V
740  PRINT
750  NEXT U
760  IF P$ = "Y" AND K$ = "Y" THEN
      PRINT : PRINT D$;"PR#0"
770  IF K$ = "Y" THEN 870
780  PRINT "ARE THESE THE CORRECT
      EQUATIONS          (Y OR N)?
      ";: GET K$: IF K$ = "Y" AND
     P$ = "Y" THEN  GOTO 570
790  IF K$ = "Y" THEN 870
800  PRINT
810  PRINT : PRINT "WHICH ENTRY I
     S INCORRECT?                I
     NPUT (EQU. # AND VAR. #).";:
      INPUT "";U,V
820  IF U > M OR V > N + 1 OR U <
     1 OR V < 1 THEN  PRINT "TRY
     AGAIN.": FOR T = 1 TO 4000: NEXT
     T: GOTO 580
830  PRINT
840  PRINT "M(";U;",";V;") IS CUR
     RENTLY ";M(U,V)
850  PRINT
860  PRINT "WHAT IS THE CORRECT V
     ALUE FOR            M(";U;","
     ;V;")?";: INPUT " ";M(U,V): GOTO
     580
870  HOME : PRINT "
      COMPUTING....":A = 0
880  GOSUB 1570
890 I = 0:J = 0
900 I = I + 1
910 J = J + 1
920  GOSUB 1850
930  IF FLAG = 0 THEN I = I - 1: GOTO
     1070
940 L = M(I,J)
950  FOR S = 1 TO N + 1
960 M(I,S) = M(I,S) / L
970  NEXT S
980  FOR R = 1 TO M
990  IF R = I OR M(R,J) = 0 THEN
     1060
1000  FOR S = 1 TO N + 1
1010 TEMP(S) = M(R,S) - M(R,J) *
     M(I,S)
1020  NEXT S
1030  FOR V = 1 TO N + 1
1040 M(R,V) = TEMP(V)
1050  NEXT V
1060  NEXT R
1070  IF M >  = I + 1 AND N >  =
     J + 1 THEN 900
1080 A = 0
1090  GOSUB 1570
1100  HOME
1110  PRINT "THE ARGUMENT MATRIX
     IN ROW ECHELON FORM IS:": PRINT
1120  GOSUB 2010
1130  IF P$ = "Y" THEN  PRINT : PRINT
     D$;"PR#"SLN
1140  IF M - A <  > N THEN 1240
1150  PRINT "THE SOLUTION TO THE
     SYSTEM OF EQUATIONS IS:"
```

組合線性方程式解答系統

```
1160  FOR T = 1 TO N
1170  IF P$ = "Y" THEN 1200
1180 CV =  PEEK (37)
1190  IF CV > 20 THEN  GOSUB 2120

1200  PRINT
1210  PRINT "X";T;"= ";M(T,N + 1)
  / M(T,T)
1220  NEXT T
1230  GOTO 1500
1240  PRINT "THE SOLUTION TO THE
 SYSTEM OF EQUATIONS IS:"
1250  PRINT
1260 B = 0
1270  FOR T = 1 TO N
1280  IF P$ = "Y" THEN 1310
1290 CV =  PEEK (37)
1300  IF CV > 20 THEN  GOSUB 2120

1310 B = B + 1
1320  IF B > N THEN 1500
1330  IF M(T,B) = 0 THEN 1420
1340  PRINT "X";B;"= ";M(T,N + 1)
  / M(T,B);
1350  FOR S = B + 1 TO N
1360  IF P$ = "Y" THEN 1390
1370 CV =  PEEK (37)
1380  IF CV > 20 THEN  GOSUB 2120

1390  IF S > N THEN 1500
1400  PRINT " -";M(T,S) / M(T,B);
 "*X";S;
1410  GOTO 1460
1420  IF B > N THEN 1500
1430  PRINT "X";B;"= X";B
1440 T = T - 1
1450  GOTO 1480
1460  NEXT S
1470  PRINT
1480  PRINT
1490  NEXT T
1500  IF P$ = "Y" THEN  PRINT : PRINT
 D$;"PR#0"
1510  IF P$ <  > "Y" THEN  PRINT
 : PRINT : PRINT "WOULD YOU L
 IKE THE ANSWER DISPLAYED
  AGAIN (Y OR N)? ";: GET K$:
  IF K$ = "Y" THEN 1100
1520  PRINT : PRINT : PRINT "WOUL
 D YOU LIKE TO SOLVE ANOTHER
 SYSTEM  OF EQUATIONS (Y OR N
 )? ";: GET K$: IF K$ = "Y" THEN
  CLEAR : GOTO 250
1530  END
1550  REM  **** END OF MAIN PROGR
 AM
1570  REM  ZERO ROW CHECKING SUBR
 OUTINE
1580  REM
1590  FOR I = 1 TO M
1600 FLAG = 0
1610  FOR J = 1 TO N + 1
1620  IF M(I,J) <  > 0 AND FLAG =
 0 THEN FLAG = 1
1630  NEXT J
1640  IF FLAG = 1 THEN 1710
1650  FOR S = 1 TO N + 1
1660 TEMP(S) = M(M - A,S)
1670 M(M - A,S) = 0
1680 M(I,S) = TEMP(S)
1690  NEXT S
1700 A = A + 1
1710  NEXT I
1720  REM
1730  REM  OVER-DETERMINED SYSTEM
 SUBROUTINE
1740  REM
1750  FOR I = 1 TO M
1760 FLAG = 0
1770  FOR J = 1 TO N
1780  IF M(I,J) <  > 0 AND FLAG =
 0 THEN FLAG = 1
1790  NEXT J
1800  IF FLAG = 0 AND M(I,N + 1) <
  > 0 AND P$ = "Y" THEN  PRINT
 : PRINT D$;"PR#"SLN: PRINT :
 PRINT "THIS SYSTEM OF EQUAT
 IONS IS OVER-DETERMINED.": PRINT
 : PRINT D$;"PR#0": GOTO 1520

1810  IF FLAG = 0 AND M(I,N + 1) <
  > 0 THEN  PRINT : PRINT "TH
 IS SYSTEM OF EQUATIONS IS OV
 ER-     DETERMINED.": GOTO
 1520
1820  NEXT I
1830  RETURN
1840  REM
1850  REM  EQUATION SWITCHING SUB
 ROUTINE
1860  REM
1870  IF M(I,J) <  > 0 THEN 1950
1880  FOR T = I + 1 TO M
1890  IF M(T,J) = 0 THEN 1970
1900  FOR R = 1 TO N + 1
1910 TEMP(R) = M(I,R)
1920 M(I,R) = M(T,R)
1930 M(T,R) = TEMP(R)
1940  NEXT R
1950 FLAG = 1
1960  GOTO 1990
1970 FLAG = 0
1980  NEXT T
1990  RETURN
2010  REM  ARGUMENT MATRIX PRINTI
 NG SUBROUTINE
2020  REM
2030  FOR U = 1 TO M
2040  FOR V = 1 TO N + 1
2050 CV =  PEEK (37)
2060  IF CV > 20 THEN  GOSUB 2120

2070  PRINT M(U,V),
2080  NEXT V
2090  PRINT
2100  PRINT
2110  NEXT U
2120  PRINT "PRESS ANY KEY TO CON
 TINUE. ";: GET K$: HOME
2130  RETURN
```

磁碟保護器－COPY VTOC法

朱永昌

相信很多朋友都有這樣的經驗，就是在SAVE程式時誤按RESET 鍵，又或是磁碟使用太久，因而出現毛病。由於檔案的索引及存取都要透過TRACK 17（詳細過程可參閱電腦時代創刊號），所以毛病往往出現在這兒，例如VTOC被破壞，又或者索引SECTOR 被毀，因而導致FILES遺失，甚至無法CATALOG；當然，這些毛病通常可以用一些工具程式加以修補，但往往要花上好幾小時翻查資料，而且修好後也不是這麼保險，最佳的方法當然是留有一份COPY，但要每一張磁碟都複製多一份，非但花費時間，而且也浪費金錢，相信沒有一位朋友肯這樣，爲了解決此問題，筆者因而寫成了此VTOC DUPLICATOR程式。

記得APPLE 的磁碟是留有磁軌 0 ，1及2用來存放DOS映像，不過DOS並非完全使用這48個SECTOR，其中二號磁軌SECTOR $5－$F是空着的，但VTOC反映給APPLE 却是已使用，所以一張磁碟實際有11個SECTOR浪費，VTOC DUPLICATOR 這程序就是將VTOC及索引SECTOR $6－$F 複製至磁軌2的空SECTOR內，以便當磁軌17被意外破壞後能迅速修補；雖然NIBBLES AW－AY或ZAP等工具程式也可担任相同的工作，但要使用它們往往要讀寫十多次，很費時間；相反這程式主要部份爲機械語言寫成，而全部工作時間僅數秒鐘，並且可以連續使用而無需重新輸入參數，十分方便；此外，讀者更可以將此程式加在HELLO 程式上，（祇要行260及280，並加上CA－LL 813 ）便能自動記下最新的資料，希望讀者能好好利用它。祝各位的磁碟身體健康。■

```
 10  REM   '######################'
 20  REM   '   VTOC DUPLICATOR    '
 30  REM   '     BY  PAT CHU      '
 33  REM   ' COPYRIGHT(C) 1985    '
 35  REM   '          BY          '
 37  REM   '    COMPUTING AGE     '
 40  REM   '######################'
 50  CLEAR : TEXT : HOME
 55 C$ = "SLOT  NUMBER->":D$ = "DR
     IVE NUMBER->":SD$ =  CHR$ (7
     )
 60  HTAB 12: INVERSE : PRINT "VTO
     C DUPLICATOR": NORMAL
 70  PRINT : PRINT : PRINT : HTAB
     4: SPEED= 100: PRINT "THIS P
     ROGRAMME WILL DUPLICATE VOLU
     ME": PRINT
 80  PRINT "TABLE OF CONTENT AND T
     RACK 11 SECTOR ": PRINT
 90  PRINT "$6-$F TO TRACK 2."
 100  SPEED= 255: PRINT : HTAB 16:
      FLASH : PRINT "WARNING" CHR$
      (7): NORMAL : PRINT
 120  PRINT "   SECTOR $6-$F OF TR
      ACK 2 SHOULD NOT": PRINT
 140  PRINT "HAVE ANY DATA OTHERWI
      SE SOME PROGRAMMES": PRINT
 160  PRINT "WILL BE ";: INVERSE :
       PRINT "DESTROYED";: NORMAL
      : PRINT "."
 180  PRINT : PRINT : PRINT : HTAB
      6: PRINT "< HIT ANY KEY TO C
      ONTINUE >";
 200  GET Z$
 210  POKE 34,9: SPEED= 120: VTAB
      23: FOR H = 0 TO 12: PRINT :
       NEXT : SPEED= 255: VTAB 1: HTAB
      12: FLASH : PRINT "VTOC DUPL
      ICATOR": NORMAL
 220  FOR A = 1 TO 14: VTAB 12: HTAB
      4: PRINT  RIGHT$ (C$,A);: NEXT
      : GET SL$: INVERSE : PRINT S
      L$: NORMAL : IF SL$ < "1" OR
      SL$ > "7" THEN 220
 240  FOR A = 1 TO 14: VTAB 14: HTAB
      4: PRINT  RIGHT$ (D$,A);: NEXT
      : GET DR$: INVERSE : PRINT D
      R$: NORMAL : IF DR$ <  > "1"
       AND DR$ <  > "2" THEN 240
 250  PRINT : PRINT "   ARE YOU RE
      ADY <Y/N>";: GET Z$: IF Z$ <
       > "Y" THEN 210
 260  RESTORE : FOR AD = 768 TO 84
      2: READ I: POKE AD,I: NEXT
 280  DATA   169,3,160,8,32,217,3,
      96,1,96,1,0,17,0,25,3,0,37,0
      ,0,1,0,0,96,1,0,1,216,239,14
      1,12,3,141,20,3,169,37,141,1
      7,3,41,15,141,13,3,32,0,3,17
      3,21,3,176,17,238,17,3,173,1
      7,3,41,15,208,235,169,2,205,
      20,3,208,215,169,0,133,72,96
 300  POKE 777, VAL (SL$) * 16: POKE
      778, VAL (DR$): CALL 813
 320  GOSUB 400
 340  PRINT : PRINT "   ANOTHER DI
      SK (Y/N)";: GET Z$: IF Z$ <
       > "Y" THEN  TEXT : VTAB 23:
      LIST 600: END
 360  POKE 34,16: HOME : GOTO 260
 400 ERR =  PEEK (789): PRINT : PRINT
 420  IF ERR = 16 THEN  PRINT SD$;
      "WRITE PROTECTED"
 460  IF ERR = 64 THEN  PRINT SD$;
      "I/O ERROR"
 500  RETURN
 600  REM
```



DISK MUNCHER的妙用

幾個熱門遊戲的抄法

一講到DISK MUNCHER，相信凡是電腦的用家都不會不熟悉。首先說說它的歷史，最早推出就是DISK MUNCH—ER 1.0 ，在當時來說可算是最快的COPY PROGRAM，抄碟只需25秒。繼而推出就是DISK MUNCHER 1.1，而DISK MUNCHER 1.1 具有VERIFY的功能，但反觀之，它的用途和功力都比不上DISK MUNCHER 1.0 。跟着DISK MUNCHER 2.0 便面世了，而功能方面，也有很大的躍進，DISK MUNCHER 2.0允許使用者更改ADDRESS MARK，而且可以任使用者選擇拷貝第＄0至＄40任何磁軌。後來，一些使用者把DISK MUNCHER改良，推出了DI—SK MUNCHER 3.0與3.3，但在功能上和2.0差不遠，而DISK MUNCHER 3.3只是較2.0多了一項TRACK INCREMENT和允許改變INSERT MARK而已。其後DI—SK MUNCHER 5.0也面世了，在功能上也沒有甚麽大的進展，但是和3.3相比，5.0比3.3好用，因爲3.3更改AD—DRESS MARK和INSERT MARK都很麻煩，而5.0比3.3整齊和簡潔得多，用來很方便。

以下都是一些程式，只要把DISK MUNCHER 的參數略爲修改，便可拷貝：

THE INSTITUTE和LUCIFER'S REALM
SIDE 1：＄0—＄23一定要抄＄23
SIDE 2：＄0—＄22

TRANSYLVANIA和THE COVETED MIRROR：
SIDE1和SIDE 2：＄0—＄22 TRACK INCREMENT→2
拷貝成功後，跟着
＄1—＄22
ADDRESS MARK→D4AA96
TRACK INCREMENT→2

BEYOND CASTLE WOLFENSTEIN：用DISK MUNCH—ER 5.0來COPY
＄0—＄2→不用改參數
＄3—＄22→ADDRESS MARK：
D5DA96
DATA MARK→D5DA

SKYFOX和ARCHON及MURDER ON THE ZINDER—NEUF抄法，都曾在電腦時代裏刊出，但我覺得他們所講的方法太麻煩，須要時間也太長，而且對於擁有一部磁碟機的使用者來說，可算是無用武之地，不過以上所講的三隻GAMES都可以用DISK MUNCHER 5.0來COPY的。

SKYFOX的抄法
＄0—＄2 不用更改參數
＄5—＄6
＄3—＄4
＄7—＄22 把DATA MARK改爲D5BB，這樣便可直接拷貝。

而ARCHON及MURDER ON THE ZINDERNEUF 抄法和SKYFOX 的抄法一樣，都可以用上述的方法去COPY，不過COPY完畢後，要利用E.D.D的帮助。先講MURD—ER ON THE ZINDERNEUF，要用E.D.D. 拷貝第＄5.25至＄6.25，而ARCHON則拷貝第＄5.25至＄5.25，這樣便可省回不少的時間。

編者註：本文作者狂龍君在文中所提及的幾個軟件拷貝方法是可行的。若然閣下未能成功，可能因爲原來軟件版本問題，與本文無關，特此聲明。■

專業電腦中心，提供中學會考電腦科補習

正在修讀電腦科的同學有沒有在學習上遇上困難，或者跟不上課程需要。

中學會考電腦科課程範圍非常廣泛，由電腦的基本結構，工作原理，程式編寫，到電腦與社會及人類的關係都有涉及。因此，在短短的課堂時間裏所能夠吸收的電腦知識，理論及經驗都極之有限，而電腦科在中學會考課程所佔的比重，及電腦科技在未來社會的地位，都很重要。有見及此，專業電腦中心特別安排一項，以實習爲重點，電腦理論爲輔的中學會考電腦科補習課程，讓學生在課餘有更長時間，更多機會實習及了解電腦的理論及知識，從而得到理想成績，爲畢業後工作作更好的準備。

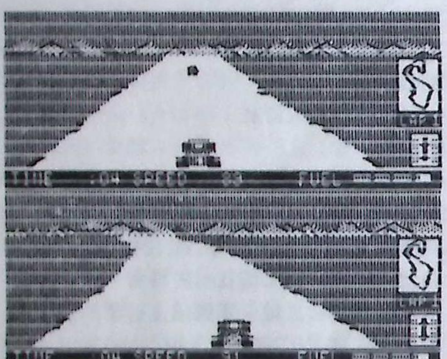
「專業電腦中心」位於油蔴地彌敦道486號益南華厦三樓，地方寬敞，交通方便，設有兩個獨立課室，一個爲上課用，另一爲實習用。實習室內設有十餘套電腦，包括完整的外圍裝置，盡量做到每人一機，加速學習進度及維持良好學習氣氛。

他們聘用的師資均爲大學及理工畢業，或現職電腦工程師，極富經驗心得及資歷。歡迎同學到校參觀報名。■



最新遊戲軟件評價　李德華

PITSTOP II



遊戲名稱：PITSTOP II
遊戲形式：賽車
評　　分：＊＊

PITSTOP II是繼FORMULA I及FORMULA II 後推出的另一賽車遊戲。PITSTOP II已把FORMULA I及FORMULA II 的缺點改進。首先是把FORMUL A I 的虛線畫面及轉彎時的畫面改進，第二是把FORMULA II 的低解像畫面轉爲高解像畫面，而新的改進是能夠同時提供兩人比賽，更加上有不同的跑道選擇。

遊戲開始時，電腦會問不同的問題，當回答所有問題後，比賽便正式開始。玩者要避過前面的賽車，直至巡廻一圈爲止，在畫面左面的跑車是顯示玩者的賽車的機件狀況，從中可得知玩者的賽車還有多少壽命。通常跑完二至三圈，玩者的跑車由於機件故障而自我毀滅，所以又要重新玩過了。■

油漆佬續集

MINER II是繼MINER I 後的另一「油漆佬」的遊戲，MINER II比 MI－NER I 更難玩，但每個LEVEL 已可看出作者是下過苦功的，但最美中不足的是沒有LEVEL 選擇，這比MINER I似乎較爲遜色了，但大致上也不失爲一個好玩的ACTION GAME 。

遊戲開始時，依然是油漆，但加上不同的食物作大力丸用途，最特別的是是加上時光隧道，令遊戲增添不少色彩。整個遊戲共有十個LEVEL ，至於能否到達LEVEL 10，就要看你自己本人了。

遊戲名稱：MINER II
遊戲類型：ACTION GAME
評　　分：兩粒半星＊＊＊■

米老奇鼠星際探險

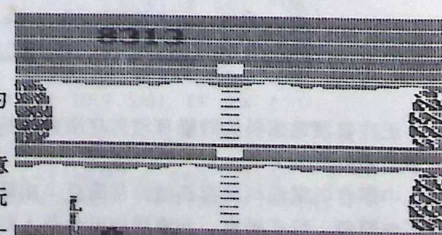
彭郁葱

另一個遊戲叫做米奇老鼠，相信大家都很熟悉這個人物吧！這個遊戲便是要你和米奇與高非狗一起到太空，進行一個偉大的任務：將分別放在九大行星的水晶收集，米奇他擁有一部十分先進的太空船，可以任意的飛往九大行星，筆者在這裏向大家提供控制太空船的密碼，大家應抄起後才玩。O 等於橙色鍵，B 等於藍色鍵。水星（MERCURY）—O。金星（VENUS）—B。地球（EARTH）—O，1B。火星（MARS）2O，1B。木星（JUPITER）—O，2B。土星（SATURN）2O，2B。天王星（URANUS）3O，2B。海王星（NEPTUNE）2O，3B　。冥王星（PLUTO）3O，3B。輸入時要注意先按橙色，後按藍色，否則電腦不會接受。而且在太空船的控制室中有一部電腦是可以帮助你的，他的名字叫XL 3O。他可以提醒你很多東西，好好把握吧！這遊戲共有四面磁碟，售價也不平宜，因此並不能大大的流行。但這個歷險故事仍是值得大家玩的。這遊戲的難度不太高，相信大家都能把握以上的方法。■



PITFALL II

PITFALL II 是一個海底尋寶式的遊戲，它和「街機」有些相似的地方，整個遊戲只是去尋金，並沒有其他玩意，而途中有不同的動物阻止你前進，玩者可技巧性地避過它們的侵襲，雖然一PITFALL II 不是一個好玩意，但它的GRAPHICS 顯示值得一讚。這遊戲最令筆者討厭的是無論玩者在那地方受到襲擊而身亡，立刻會回到終點，而一路上不停地扣分，分分鐘會一分都冇。■



遊戲名稱：PITFALL II
遊戲類型：海底尋寶
評　　分：＊

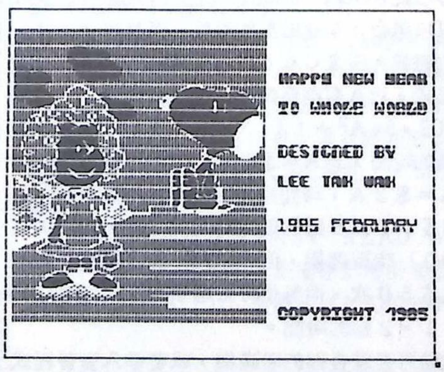
花生砌圖

很多小朋友都很喜歡看花生漫畫，今次介紹的是非常適合五歲至八歲的小童玩的PEANUTS PUZZLERS。

當孩子第一次望到這個砌圖遊戲，一定會被那些構圖及聲响所吸引，大人可從旁指導兒童應怎樣把那些圖像連接，當那些圖像連接好後，那幅畫便會變成一幅非常生動的圖畫，而且更顯示出所用的時間，兒童見到一定會笑逐顏開，大人更可把時間記錄下來，便可訓練兒童的記憶力了。

整個PEANUTS PUZZLERS 共有八種不同的砌圖，是一個非常值得推薦的兒童遊戲。■

遊戲名稱：PEANUTS PUZZLERS
遊戲類型：砌圖遊戲
評　　分：＊＊＊



UP'n DOWN

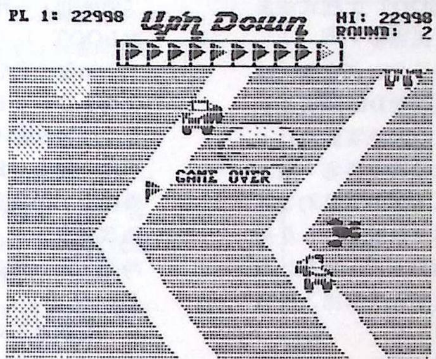
UP'N DOWN 是一個非常刺激的爬山車遊戲。玩這遊戲必須有良好的駕駛技術。

在遊戲開始時，玩者會在一些頗窄的山路上，把JOYSTICK 推上，爬山車就會發動。在山路上，玩者會遇到不同的旗桿，只要碰上便能拿到，在畫面上方會顯示玩者取了多少旗桿及還有多少旗桿未取，當玩者把所有旗桿取後便能升LEVEL。

在遊戲途中，有不同的爬山車阻止你前進，你可按BUTTON 0或BUT—TON 1 跳起避過，當跳起的時候，你更可把控制桿推上推落來控制爬山車的彈跳力，如力度恰當，更可把阻撓你的過山車毀滅。

在不同的LEVEL 當中，會有不同的難關，例如掘頭路、殭車、鏟泥車等等。而每升一次LEVEL ，便會有不同的獎分，筆者最多只能拿到三千分，不知你們又如何呢？■

遊戲名稱：UP'N DOWN
遊戲類型：ACTION
評　　分：兩粒半星 ＊＊＊



捉鬼敢死隊

彭郁葱

在聖誕節時，有一套電影叫「捉鬼敢死隊」，主角手執核能槍，version威風的。現在你不需要羡慕他，因爲你也能大展身手了。

遊戲開始了，現出「捉鬼公司」的標誌，然後奏出主題曲，而且還有歌詞，可說是聲色俱全。跟着輸入名字，電腦會問你銀行有存款嗎？有就鍵入客戶號碼，否則鍵入「N」，如果沒有存款，銀行會借一萬元給你。（好好利用吧，因爲這是要還的。）首先去買車；有福士車、日本車，及美國車，還有一架先進的太空車，因爲筆者銀行沒有存款故此不能買到，因爲太空車價值一萬五千元。買了車後便選擇武器，計有：窺鬼器、核能槍、捕鬼器等。買齊東西後，可鍵入「E」便可開始行動了。

螢光幕可看見有四隻小鬼，不要看輕他們，因爲那小鬼結合，會成爲那賣棉花糖的人，他的破壞力是無可估計的，相信大家都記憶猶新。

大家要注意那些樓宇，如果是微微的閃動那便是鬼屋了。用那襟章編好了位置後，你的車便會駛到目的地，開始捉鬼行動了。

首先放下捕鬼器，然後兩人一起發出核能死光，不讓鬼怪逃走，等到鬼怪到了捕鬼器的上空時，按下「BOTTO—N 1」，捕鬼器中便會射出捕鬼光，把鬼怪拉進捕鬼器中。由於你的電力不足，故此你只有一次機會，錯過了便會遭受鬼怪的襲擊。

如果你沒有空的捕鬼器便駛車回總部，放下鬼怪，便可再用了。■

遊戲名稱：GHOSTBUSTER
遊戲類型：ACTION GAME
評　　分：＊＊＊＊



TAPPER

你是一位酒吧的老闆，現在有很多的酒客光顧，現請你以最快速度把啤酒送給每位客人，現在開始吧！

TAPPER 是一個非常好玩的考反應遊戲，玩者只要把啤酒遞給酒客便行，但一些酒客往往一杯啤酒不足夠，此時你便要立刻把空的啤酒杯接住，然後再裝滿另一杯遞給酒客，直至他們飲醉便行，在遊戲期間，如果玩者在枱面上拿到貼士，便有一場歌舞表演，那些酒客便會靜靜欣賞，那可拖延時間，當然當歌舞過後，那些酒客又會大吵大嚷地要求快些給他們啤酒了。

過了三版後，便會有一版BONUS 的LEVEL ，用者只要估中那隻酒杯，便會有三千分獎分。

第四LEVEL 是改變了少許形式，而且櫃枱的編排是由右至左及加多些人，筆者爲了趕及截稿時間，只玩到LE—VEL 6 ，聽講可以玩到太空LEVEL ，不知是否屬實。■

BOOT 0 程式的活用與分析

莫信

相信大家都知道，沒有BOOT　0程式及類似的程式，我們不可能啓動一部磁碟機並進行BOOT　DOS 的工作，自從DOS3·3版本面世後，現今沿用的BOOT 0 程式並未受到改變，最簡單的原因是它合用，並且優點多。

到底BOOT　0程式是怎樣的一個樣子？它負責將一個對映表放於位址＄36C至＄3D5，並啓動DRIVE 1，設定讀取資料狀態，將磁頭探臂重定歸位（移至零號軌），設定緩衝區開始位址爲＄800，要讀的區與軌爲零，隨之開始讀取磁碟第0軌的資料，搜尋第0區，將第0區的342個資料讀入至位址＄300至＄355及位址＄800至＄8FF，然後比較CHECKSUM，若讀取正確，則重組資料碼（因DOS 3·3是採用“6及2”分塊寫入法來寫入資料碼的）；否則回返，重新讀取。

重組“6及2”碼後，增加緩衝區開始位址，要讀取區，並要由讀取區與位址＄800比較，決定是否繼續讀取下一區，若BOOT 0工序全部完成，則JMP＄801進行BOOT1的動作。

似乎BOOT　0程式的工作情況大致上我們已經了解了，但你們可曾留意過相同的BOOT　0程式在任何一個槽位都可以如常工作，其實是因爲設計BOOT　0程式的程式師玩了一個魔術，而筆者對這個魔術認爲極精彩，所以特別提出討論，觀看附文程式，在位址＄C621處，程式JSR＄FF58，而位址＄FF58是存放著內含值＄60，亦即是組合語言RETURN 的意思，那麼JSR＄FF58豈不是多餘，其實並不然，它的主要作用是配合位址＄C624的指令TSX將STACK　POINTER的內含值取出並存放於X暫存器內，而LDA＄100，X便是將STACK POINTER 的指標存放於A累積器內，如此我們便可以知道到底由那個槽號位置呼叫磁碟機作用了，若我們用IN＃6、PR＃6、C600G或6CTRL－P來呼叫槽號6的磁碟機作用，那麼A累積器內便存有值＄C6，經過4次的ASL（ACCUMULATOR　SHIFT LEFT），A累積器內含值便變成＄6.0，再將A累積器存入X暫存器，如此便可知道應該啓動那個槽號的DRIVE　1了，同樣地，若以C500G呼叫，得出來的答案便變成＄50了。

減低BOOT機時的噪音

每次我們使用BOOT　0程式做BOOT　DOS 的動作時，我們都會聽到不大不小的振動聲音，原因是磁頭歸位（第0號磁軌），APPLE磁碟機是以一片塑膠片來防止磁頭跑出第0號軌以外的範圍，當磁頭到達邊界時，探臂便與膠片碰撞，那些振動聲音便是因碰撞而造成的。但聽到這種聲音總令人感到不舒暢，又磨損一些了，其實就算我們用金屬片來防止磁頭跑出第0號軌以外的範圍，也是會受到磨損的，“他媽的，不知那個龜孫子寫的BOOT　0程式？搞得我每次BOOT　DOS都心裏不舒服。”相信你們也有這種想法吧。不如改了它吧？改了它，改那處呢？其實很簡單，祇要將位址＄C64B的本來値＄56改爲＄2A就成了，我並不是說你將永遠不會聽不到那令人心悸的振動聲，但這個改動可以令你每次BOOT　DOS時安心一些，通常那振動聲都變得極小，因碰撞減少了，而且你若將位址＄C64B的內含値改爲＄2A，它可以減少你1·2秒的BOOT　DOS 的時間，換一句話說，你可以省回1·2秒的時間，原因爲何？

我們並不可以說BOOT　0程式不好，因爲這個BOOT 0程式並不祇可以適合35軌磁機的用戶，而且可以適合40軌或更多軌的磁機。在＄C64C處JSR＄FCA8是時間延遲常規副程式，以A暫存爲時間單位，我們可以用公式：時間延遲＝（$2\cdot5*A^2+13\cdot5*A+13$）時脈週期來計算每次延遲的時間，若A＝＄56，時間延遲＝19664時脈週期；若A＝＄2A，時間延遲＝4990 時脈週期。如此我們每次使用＄FCA8處的延遲時間程式時，便減少（19664－4990）時脈週期，但BOOT　0程式總共使用延遲時間程式共＄50次，而每個時脈週期＝1·023US，所以總共節省了1·2秒的時間。

測試這個改動可否適合你的磁碟機，祇要進入監督程式，以＊8600<C600，C6FFM，將位址＄C600至＄C6FF的程式碼搬移至＄8600至＄86FF，現在若你們用8600G來BOOT　DOS是一樣可以成功的，在未8600G之前，我們將位址＄864B的內含値改爲＄2A，然後寫一個簡短的程式：

```
4000-   A9 27        LDA   #$27
4002-   8D EC B7     STA   $B7EC
4005-   A9 00        LDA   #$00
4007-   8D F4 B7     STA   $B7F4
400A-   A9 B7        LDA   #$B7
400C-   A0 E8        LDY   #$E8
400E-   20 B5 B7     JSR   $B7B5
4011-   60           RTS
4012-   00           BRK
```

這個程式將探臂移至軌道＄27（即第40軌），若你使用的是祇有35軌的磁碟機，請將＄4001的值＄27改

BOOT 0 程式的活用與分析

爲＄２３，當用＄４０００G執行了這個程式後，探臂便與軌號零成爲最遠的距離，再以＄８６００G執行那個被搬移與改動的BOOT ０程式，成功了是嗎，並且聽不到那些心悸的振動聲，是嗎？若果你不能成功的BOOT DOS，那麼就將＄８６４B的內含值加大，直至成功BOOT DOS爲止吧。

不過若你們的磁碟機不祇使用４０軌，那麼位址＄C６４B的內含值便要由＄２A加大，不過在市面上所買到的，絕大部份是３５軌與４０軌，所以用＄２A是足夠應付的。

至於如何將硬件改動，請教專家吧，在這裏筆者祇提供你一個方法。

快速直接讀取任何一個SECTOR

至於活用方面，我們可以利用BOOT ０程式的其中一部份，配合位於＄B９A０處的SEEKABS常式，寫一個讀取一個SECTOR的程式，它比直接使用RWTS副程式還要快一倍以上的時間，但你要留意的是它所讀的並不是DOS3·3段落，而是實際磁碟的段落，段落交錯對映表是在圖(一)，你們可以拿來參考。

若使用這個程式祇需約０·４秒便可讀取一個SECTOR（以不移動探臂計算），以RWTS副程式直接讀取一個同軌的SECTOR，則需要約１·２秒的時間，快得多，就算原來探臂是在軌４０號，而我們要讀取軌０區０號，則需時約０·７秒，用RWTS則需要１·５秒，所以快慢差距極大。

現在就動手將這個FAST READ A SECTOR存入電腦，試試它的威力吧。祇要你們將所要搜尋的軌與區存於位址＄１C及＄１D，而緩衝區起始位址於＄１E及＄１F（以低位址先存入），然後以＄５０００G便成了。舉一個例子，現我們要將第４號軌第７磁區存入於位址於＄７９００處，那麼我們祇要打入：

```
* 1C :  4   07    00  79   N   5000G
        ↑   ↑      ︸           ↑
        軌  區   緩衝區起始位     執行命令
```

便可以得到預期的效果。■

圖一　段落交錯的對映表

實際的磁碟段落	DOS 3·3 段落
0	0
1	D
2	B
3	9
4	7
5	5
6	3
7	1
8	E
9	C
A	A
B	8
C	6
D	4
E	2
F	F

```
*********************
** READ A SECTOR **
**   A$5000,L$FB   **
*********************
5000- A2 20 A0 00 A2 03 86 3C
5008- 8A 0A 24 3C F0 10 05 3C
5010- 49 FF 29 7E B0 08 4A D0
5018- FB 98 9D 56 03 C8 E8 10
5020- E5 AD EE C0 AD EC C0 AD
5028- EA C0 AD E9 C0 AD EC B7
5030- 0A 8D 78 04 A2 60 A5 1C
5038- 0A 20 A0 B9 18 AD 78 04
5040- 6A 8D EC B7 AD 00 08 85
5048- 07 AD 01 08 85 08 18 A5
5050- 1D 69 01 8D 00 08 A9 60
5058- 8D 01 08 A5 1E 85 26 A5
5060- 1D 85 3D A5 1C 85 41 A5
5068- 1F 85 27 20 5C C6 A5 07
5070- 8D 00 08 A5 08 8D 01 08
5078- AD EA C0 AD E8 C0 60 00
5080- 00 00 00 00 00 00 00 4C
5088- 00 50 00 00 00 00 00 00
5090- 00 A5 41 00 00 00 00 00
5098- 00 00 00 00 00 00 00 00
50A0- 00 00 00 00 00 00 00 00
50A8- 00 00 00 00 00 00 00 00
50B0- 00 00 00 00 00 00 00 00
50B8- 00 00 00 00 00 00 00 00
50C0- 00 00 00 00 00 00 00 00
50C8- 00 00 00 00 00 00 00 00
50D0- 00 00 00 00 00 00 00 00
50D8- 00 00 00 00 00 00 00 00
50E0- 00 00 00 00 00 00 00 00
50E8- 00 00 00 00 00 00 00 00
50F0- 00 00 00 00 00 00 00 00
50F8- 00 00 00 00 00
```
■

各期程式磁碟磁帶總目錄

1
1 突破傳統的鎖盤
2 撞球遊戲
3 勇闖電腦室
4 迷宮救美
5 將 APPLESOFT 變爲 FORTRAN
6 上、下、左、右移動文字畫面
7 測試工具— BOOLEAN 程式
8 騰空更多磁碟空間 —搬移HELLO
9 重現刪除之檔案
10 電腦時代月曆
11 COMPUTEXT 文字處理
12 時日相距計算程式
13 增強英文字滙能力
14 幼童數學四則運算練習
15 過四關

2
1 貨櫃處理
2 沙蟹幻方局
3 巧破密令
4 美術字繪劃器
5 小階字母處理
6 程式轉載器
7 彩色電視／監視器 色彩測試
8 保留遊戲積分記錄
9 數學套件程式
10 磁碟修護工具
11 加送電腦時代月曆 圖像一個

3
1 DOS 的自動同步追踪器
2 動量不滅定律模擬法
3 自製磁碟索引表封套
4 蘋果編曲器
5 幼童認數字初階
6 賽跑遊戲
7 步步追踪
8 DOS KILLER
9 圍棋遊戲
10 巴哈的古典樂章名作
11 倒轉螢幕畫面
12 雜誌目錄 分類歸檔系統
13 神經病醫生—神經鬆弛體操

4
(1) DOS 操作指令簡化器
(2) TEXT FILE 有些甚麼內容？
(3) 立體迷宮
(4) 英文串字練習
(5) 如何以機械語言來修補 DOS
(6) 電腦動畫製作原理。
(7) 豐富設計的學前兒童 教育程式
(8) 應用數學套件程式(二)
(9) DOS WRITER
(10) 穿梭機哥倫比亞號機
(11) 找回失落了的 STEP 及 TRACE
(12) TRACER控制常式
(13) 將 RAM 卡作爲磁碟機使用
(14) 煮房計時器
(15) 太陽系模擬系統

5
1 私人約會備忘日曆
2 三度空間顯像系統
3 將 CP／M 檔案轉爲DOS
4 蘋果萬花筒
5 高解像度文字產生器
6 修訂DOS 操作指令簡化器
7 濃縮圖片儲存空間
8 幼童數學練習——數字咭
9 加數機大登場
10 備忘 MEMO紙
11 光速計數器
12 剩餘磁碟空 SECTOR數目
13 16進制輸入數字鍵盤
14 問題棋盤
15 神槍智破夾萬
16 隔山打牛
17 大空船降落任務

6
(1) 私人會計系統。
(2) APPLESOFT 擴展版本。
(3) GO] [HELLO 的程式
(4) APPLESOFT 偵錯系統。
(5) DOS 指令再修訂版本。
(6) 布倫時鐘。
(7) AMPERSAND 數基轉換器
(8) 完整索引錄。
(9) 砌積木。
(10) 神奇數字方塊。
(11) 亡命飛車。
(12) 迷宮搜秘。
(13) 太空怪侵略戰。
(14) 累加遊戲。
(15) 爲磁碟命名。
(16) 任意定義鍵盤。
(17) 捉蟲訓練問題。

7
(1) 會說話的蘋果
(2) 私人財政預算系統
(3) 私人收入會計系統
(4) 高解像字符繪劃器
(5) 圖像轉載器
(6) 螢幕文本編輯處理器
(7) 濃縮圖片程式修訂加強版本
(8) 蘋果杯大賽車
(9) 十六個難度的食鬼遊戲
(10) 高爾夫球遊戲。
(11) 空戰驚魂
(12) 美麗圖案設計家
(13) 更多色彩
(14) 列印的困擾
(15) 十一條空 SECTOR
(16) 捉蟲訓練
(17) 80字行咭例子程式

8
(一) APPLESOFT 滙編器
(二) 最簡單的拆帶程式
(三) 英文打字練習
(四) 編印單價LABEL
(五) 高解像頁處理工具箱
(六) 簡易快速的記憶移動方法
(七) 眞正要動腦筋的小學四則運算
(八) 七海霸王 (DEPTH CHARGE)
(九) 原子黑洞
(十) 蛇之噩夢
(十一) 收銀機
(十二) 改良的閃動訊息
(十三) 兩欄索引顯示
(十四) 自動三角函數曲綫繪劃系統
(十五) 任意鍵盤 (三個)
(十六) 每期捉蟲問題
(十七) 奇妙的 &

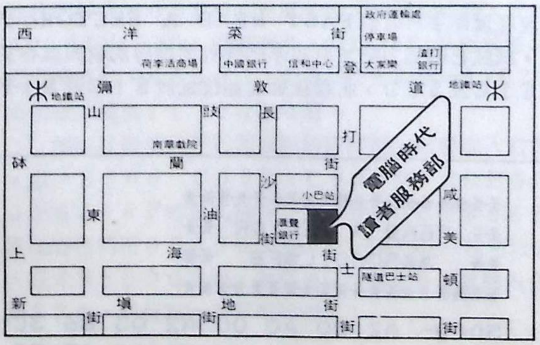
9
(1)磁碟檔案黃頁系統
(2)增強鍵盤輸入及螢幕輸出功能的系統
(3)將工具程式放入DOS佔用的
三條磁軌上
(4)快速尋找及磁碟矩陣儲存與提取
(5)新DOS命令改換程式
(6)任意設定目錄索引顯示模式
(7)捉迷藏
(8)古堡之謎
(9)高空擲物
(10)有趣的排序
(11)捉虫問題及答案
(12)第八期捉虫問題答案
(13)印出高解像數目字

10
(1) 可隨意訂定格式之資料紀錄系統
(2) 太空戰士
(3) MARCO鍵盤
(4) 射鴨仔
(5) 奇妙的&——快速分類整理程式
(6) 40軌磁碟工具程式設計
(7) 變數表顯示
(8) 跳越加速器
(9) IQ方塊
(10) 快速檔案搜索
(11) 將DOS移上RAM咭
(12) 培養兒童的記憶能力
(13) 初中代數式運算
(14) 萬用數基轉換器
(15) 十六欄機械碼編印
(16) 捉蟲問題及答案

11
(1) 家庭購物備忘及存貨紀錄系統
(2) 可隨意訂定之資料紀錄系統（下）
(3) 奇妙的＆專欄——矩陣定值
(4) 40軌磁碟工具程式之(二) 單軌 INIT
(5) 兩個自動減省程式佔用之記憶空間應用程式
(6) 對我說
(7) 趣意盎盎的歡樂棒
(8) 英文語句文法分析練習系統
(9) 一行程式示例（共七個）
(10) 美化程式列表
(11) JACKIES HEAD
(12) 魔域迷宮
(13) 巨蟹求生記
(14) 自動日子記錄
(15) 空 SEOCTOR圖表
(16) 捉虫訓練——FACTORIAL！

12
① 家庭開支及個人消費管理系統
② 電子郵箱
③ 40軌程式COPY工具
④ 諾曼第登陸戰（模擬戰爭思考遊戲）
⑤ 工兵掘地雷
⑥ 新改良蛇之噩夢
⑦ 魔域迷宮（APPLESOFT 版本）
⑧ 讓九個月大的嬰孩接觸電腦吧！
⑨ 有多項功能的AMPERSAND工具程式
⑩ 磁頭清潔器
⑪ 程式拯救者
⑫ TAB & XY
⑬ 符號逆輸型（RPN）計數機
⑭ 比RAM DISK更好的RAM咭控制中心
⑮ 圖形建造師
⑯ 捉虫訓練——SINE曲線
⑰ 一行程式

歡迎補購各期
程式磁碟或雜誌



精選程式磁碟

編號：CA00A
內容：遊戲及教育程式
遊戲程式
1、捉球遊戲（第一期）
2、勇闖電腦室（第一期）
3、迷宮救美（第一期）
4、過四關（第一期）
5、沙蟹幻方局（第二期）
6、貨櫃處理（第二期）
7、賽跑遊戲（第三期）
8、步步追踪（第三期）
9、圍棋遊戲（第三期）
10、立體迷宮（第四期）
11、穿梭機哥倫比亞號（第四期）
教育程式
12、增強英文滙能力（第一期）
13、幼童數學四則運算（第一期）
14、數學套件程式（第二期）
15、動量不滅定律模擬法（第三期）
16、幼童認數字初階（第三期）
17、豐富設計的學前兒童教育程式（第四期）
18、應用數學套件程式之二（第四期）

編號：CA00B
內容：應用程式精選
1、電腦時代月曆（第一期）
2、COMPUTEXT 文字處理（第一期）
3、美術學繪劃器（第二期）
4、彩色電視／監視器色彩測試（第二期）
5、保留遊戲積分紀錄（第三期）
6、自製磁碟索引封套（第三期）
7、APPLE 編曲器（第三期）
8、雜誌目錄分類歸檔系統（第三期）
9、DOS WRITER（第四期）
10、電腦動畫製作原理（第四期）
11、將 RAM咭作爲磁碟機使用（第四期）

編號：CA00 C
內容：私人會計系統
(1) 私人財政預算 （第七期）
(2) 私人收入紀錄 （第七期）
(3) 私人會計程式 （第六期）

編號：CA00D
內容：工具程式集
(1) APPLESOFT程式偵錯系統（等六期）
(2) 將DOS 移上RAM 咭（第十期）
(3) MACRO 鍵盤 （第十期）
(4) RAM 咭控制中心 （第十二期）
(5) 程式編輯器 （第十三期）

每期程式磁碟現仍售29.9元；
精選程式磁碟每張售價45元
（除CA00C 只售30元）。

各期程式磁碟總目錄

13

(1) 行政思考決策輔助系統——成功人士必備工具
(2) 全能標貼製作器
(3) 大力水手——POPEYE
(4) 火樹銀花賀聖誕——平安夜
(5) 低解像圖輔助繪劃器
(6) 達爾文進化論的電腦模擬實踐
(7) 程式編輯器
(8) 磁碟檔案地圖
(9) 增加DOS的CATALOG暫停功能
(10) 新游標
(11) 40軌磁碟改動器
(12) 餅形圖
(13) 自己設信紙版頭
(14) 捉虫問題及答案
(15) 一行程式得獎作品

14A

(1) 高解像圖魔術師
(2) 以軟件來達成的音響合成器
(3) SCREEN SAVER
(4) 磁碟度身定造器
(5) DOS的快速存取
(6) 一行程式得獎作品選
(7) HIDE：將程式隱藏
(8) 英文字義辨識練習
(9) 數恐龍學習加、減運算
(10) 宇宙城保衛戰（麥法基專欄）
(11) 最後一個立體打井遊戲
(12) 月球一小步，人類一大步
(13) 音量控制
(14) 將APPLESOFT程式轉爲TEXT檔案來編輯
(15) 捉虫訓練——DOS指令的執行

14B

DOS 3.3 FORMAT（SIDE A）
(A) 令蘋果發言的最佳工具MOCKING BOARD
SPEECH WRITER
(B) 本港郵費計算諮詢系統
CP／M FORMAT（SIDE B）
(C) CP／M的密碼檔案
(D) 程式設計與FORTRAN所帶來的好處
（雙精度計算例子）

15

(1) 活動電子訊息顯示板
(2) 借貸分期還款計算系統
(3) 麥法基遊戲專欄——DANGER RANGER
(4)「福」臨香江
(5) 將高解像頁倒轉
(6) 機械碼輸入編輯器
(7) 程序分解擴展器
(8) 利用印字機印出多款字形
(9) 檔案密碼鎖
(10) 兩個新DOS指令之一MREAD
(11) 電腦繪圖專欄——立體繪圖的原理示範程式
(12) 氫氧碳化學分子排列結構模擬
(13) 豬仔錢箱
(14) 3－6－9鬥智遊戲
(15) 超級越軌大賽
(16) 電腦音樂動畫——MONICA
(17) 一行程式入選作品
(18) 捉虫問題及答案

16

(1) 專業室內設計師
(2) 大忙人時間顧問
(3) 學習迷宮產生的原理及自編迷宮遊戲
(4) 使用更方便的BOOT DRIVE 2 軟件
(5) LIST大師
(6) CATALOG HELPER
(7) 40軌磁碟SECTOR使用情況顯示
(8) 磁碟保護器——COPY VTOC法
(9) 兩個新DOS指令MREAD & MWRT（下）
(10) 爲你的程式配上背景音樂
(11) 成雙成對
(12) 吃不到的葡萄不是酸的!!
(13) 動態／策略兼備的戰爭遊戲：BEACH HEAD
(14) 一行程式入選作品
(15) 捉虫問題及答案
(16) 組合線性方程式解答系統
(17) 太空侵略者封閉戰CLOSING－ALIEN

補購請到——
上海街438號同珍商業中心11字樓1104室（上海街／登打士街交界）
電話：3-312007

聰明的 RUN

呂樹偉

你有否爲FILE TYPE MISMATCH而煩惱過呢？即使是一位熟練的操作員，也會因此而煩惱。你有沒有想過用一RUN去處理二進位元檔，甚至EXEC 文字檔呢。當電腦接受了一個DOS的RUN，BRUN或EXEC時，跟着便做開檔的動作（$A2A8），檢查檔案型號是否配合，是則繼續，不是則印出FILE TYPE MISMATCH。到此，我們只要把RUN的副程式進入點改變，便可在RUN之後跳到一特定的副程式，處理各種檔案。

副程式先是呼叫檔案處理程式做開檔的動作（$A2A8），然後關閉（$A316），在此過程中已將檔案資料存入參數表中（$B5C2），然後把FILE LOCK位元清除，逐一檢查處理。副程式是放在一塊DOS未用的記憶區內（$BCDF），把副程式放在這裏有一好處，經此DOS INIT的磁碟都有些功能。程式如下：

```
BCDF- 20
BCE0- AB A2 20 16 A3 A9 7F 2D
BCE8- C2 B5 F0 0A C9 04 D0 03
BCF0- 4C 8E A3 4C D1 A4 4C C6
BCF8- A5
```

我想你們也會想到，若把BRUN，EXEC的副程式入口址也改了，也將會有同樣功能。

```
9D32:DE BC (EXEC)
9D52:DE BC (BRUN)
```

現在不要做FILE TYPE MISMATCH的奴隸了！ ■

徵求一行程式有獎遊戲

在上月我們收到許多〔一行程式〕的投稿，其中不乏精彩的作品。

這個遊戲旨在鼓勵大家動腦筋，利用最簡短的程式，設計一些有創意的效果。

今次未刊登你的作品的讀者請勿灰心，可能我們會在下一期登出。

各位投稿者請在來稿中加上程式內容說明。以前各期的得獎者請盡快前來讀者服務部領取應得的獎品。

各位讀者寫來程式時，必須用多行的形式寫來（參考本文的各位得獎例子），而且要書寫清楚。投稿請寄來CPO BOX 71193。入選者可獲得任何一期的程式磁碟一張（請註明要的磁碟）。來稿信封請勿封口。

本程式是一個用來簡化分數的程式。首先打入分子(A)和分母(B)的數值，電腦便會很快把分數化爲最簡。

吳宇光君可贏得第六期程式磁碟一張。

```
10    HOME
   :  INPUT "A/B";A,B
   :  X = A
   :  Y = B
   :  FOR I = 0 TO 1
   :  D = A - B
   :  C = B > A
   :  A = A - D * C
   :  B = B + D * C
   :  A = A - INT (A / B) * B
   :  I = A = 0
   :  NEXT
   :  PRINT X;"/";Y;"=";X / B;"/";Y
        / B
```

這是一輛正在風馳電掣地行駛着的吉普車。甘德璋君乘它來服務部取第十期程式磁碟一張。

```
10    HGR
   :  HCOLOR= 3
   :  ROT= 0
   :  SCALE= 1
   :  FOR I = 768 TO 799
   :  READ A
   :  POKE I,A
   :  NEXT
   :  POKE 232,0
   :  POKE 233,3
   :  FOR X = 10 TO 270
   :  DRAW 1 AT X,99
   :  A = PEEK ( - 16336)
   :  XDRAW 1 AT X,99
   :  NEXT
   :  RUN
   :  DATA 1,0,4,0,18,173,45,40,37,228,
        63,28,28,28,86,17,55,63,39,39,5
        5,39,55,55,53,55,45,21,45,40,5,
        0
```

本程式是一個簡單的COPY PROGRAM，能夠將DOS 由DRIVE 1 抄到DRIVE 2，若將迴路FOR T=0 TO 2 改寫成FOR T=0 TO 34 ，則可將整面碟COPY，但速度不快。（留意 ，磁碟一定要FORMAT 過）

李荔洲同學可得到一張第十三期的程式磁碟。

```
10    DATA 32,227,3,76,217,3
   :  A = 47092
   :  B = A - 8
   :  C768
   :  D = B + 1
   :  POKE A - 1,0
   :  POKE B - 1,0
   :  FOR I = 0 TO 5
   :  READ T
   :  POKE C + I,T
   :  NEXT
   :  PRINT "PRESS ANY KEY TO START"
   :  WAIT - 16384,128
   :  FOR T = 0 TO 2
   :  FOR S = 0 TO 15
   :  POKE D,S
   :  POKE B,T
   :  FOR I = 1 TO 2
   :  POKE A,I
   :  POKE B - 2,I
   :  CALL C
   :  NEXT I,S,T
   :  PRINT "DONE!"
```

本程式在執行完後，可將〝DISK VOLUME〞的字樣改變成〝COMPU—TING—AGE..〞

當然，要想改成任何字樣本程式都能辦到，只要將DATA行句的輸入字樣改變便成，相當簡單實用，李文豪君可贏得第十五期磁碟一張。

```
10    FOR J = 44480 TO 44482
   :  POKE J,234
   :  NEXT
   :  POKE 44463,15
   :  POKE 44465,171
   :  POKE 44466,179
   :  FOR I = 46010 TO 45996 STEP - 1
   :  READ A$
   :  POKE I, ASC (A$) + 128
   :  NEXT
   :  DATA C,O,M,P,U,T,I,N,G,-,A,G,E,.,
   :  PRINT CHR$ (4);"CATALOG"
```

電腦時代讀者服務部與 COMPUHOUSE 聯合主辦

APPLE FAN CLUB

APPLE FAN CLUB（以下簡稱AFC）屬於一個不牟利性質的組織，旨在聯繫蘋果電腦用家，促進大家對電腦的應用。歡迎蘋果電腦用家參加。

AFC 提供的權利包括：

① 免費借用最新出版的蘋果電腦軟件。
② 以批發價錢供應會員各類磁碟、磁碟機、週邊咭。
③ 獲得每月三份的十日刊通訊錄，通訊錄報導各方面的最新軟件出版消息，完全免費。
④ 爲讀者搜羅各類的MANUAL 。
⑤ 供給各種新軟件的BACK UP或破解方法。
⑥ 會員可以安坐家中足不出戶取得所借用的軟件。

每個月，港九的高登及國都商塲都推出有不少的新軟件，包括遊戲及工具軟件。我們將一一的搜羅這些軟件，存放在AFC內的SOFTWARE LIBRARY 內，以便供會員借用。每隔十天我們會寄出一份最新軟件出版的消息動態及名單，會員收到通訊錄後，可隨意選擇要借用的軟件，塡妥隨附的借用申請表格連同空磁碟（我們亦以批發價錢供應各類型磁碟）寄來AFC。我們收到借用表格後，在五個工作天內便會將會員要借用的軟件寄出。或者申請者可於指定時間到電腦時代讀者服務部攞取。每名會員每月可免費借用六個軟件，若要多借軟件，每個只收取象徵式一元的服務費（可用郵票代付），此外，別無其他收費。同時，我們亦提供有關的軟件的MANUAL ，我們只收回影印成本。

招收基本會員（名額有限　報名從速）

首期會員通訊將於三月一日開始寄出

參加AFC資格：只要是蘋果電腦用家即可。

收費辦法：參加者每年須繳付年費180 元。亦可分期繳交，辦法如下：

(1) 一次過繳交一年會費　＄180
(2) 分二期每半年繳交一次 ＄108

各位只要塡妥下列表格，連同劃線支票寄來九龍中央郵政信箱CPO BOX 71193，電腦時代出版社收卽可，支票抬頭塡寫〔COMPUTING AGE PUBLISHER〕，信封面請註明〔AFC〕。分期繳付會費的申請先繳付首期會費。

APPLE FAN CLUB 申請入會表格

申請者姓名：（英文）LAM LAI HO　（中文）林禮豪

通訊地址：（英文）

（中文）...A 18樓

通訊電話：... 業：學生

支票號碼... 申請日期

特惠優待證號碼＊ ______ 電腦時代訂閱戶＊ 17M506 COMPUSOFT磁碟月刊訂閱戶＊ ______

＊假如是訂閱戶請塡寫訂戶號碼

繳付會費辦法。

☑ 一次繳付　☐ 分二期繳付

NO. ____
Official Use

＊＊香港以外地區申請者需加附郵費（空郵）40元，另外每次申請借用軟件需自付郵費。
請容許十五至三十日時間辦理登記會員手續。我們會發信通知。

電腦時代程式磁碟

每張只售HK$29.90

電腦時代每期都刊登多個精彩的程式，其中必然有你喜歡的。當你讀完文章興緻勃勃的坐下來欲進一步深入了解程式的精彩處時，看見那大堆冗長的列表，就甚麼興趣也沒有了！那麼，你爲何不購買一張「電腦時代程式磁碟」呢！

每期出版的電腦時代所登載的程式，均全部紀錄入一張高品質的磁碟上，並隨雜誌同時出版。爲省卻打入程式的麻煩，徒然浪費你寶貴的時間，免除了花精神捉錯打字的痛苦，今日就請就近的特約經銷處或電腦時代讀者服務部/COMPUHOUSE購備一張。

程式磁碟每張只售港幣 29.90 元。

第16 期程式磁碟內容

*All Programs are in Apple version

(1) 專業室內設計師
(2) 大忙人時間顧問
(3) 學習迷宮產生的原理及自編迷宮遊戲
(4) 使用更方便的BOOT DRIVE 2 軟件
(5) LIST大師
(6) CATALOG HELPER
(7) 40軌磁碟SECTOR 使用情況顯示
(8) 磁碟保護器——COPY VTOC 法
(9) 兩個新DOS指令MREAD & MWRT（下）
(10) 爲你的程式配上背景音樂
(11) 成雙成對
(12) 吃不到的葡萄不是酸的 !!
(13) 動態／策略兼備的戰爭遊戲：BEACH HEAD
(14) 一行程式入選作品
(15) 捉虫問題及答案
(16) 組合線性方程式解答系統
(17) 太空侵略者封閉戰CLOSING－ALIEN

本期程式磁碟雙面灌錄　讀者留意

爲方便各區讀者購買程式磁碟，我們新增加多個特約經 銷處：

以下爲各特約經銷處地址：

中環

百老滙攝影器材有限公司（第五店）
域多利皇后街 3 號（中環街市側）
電話：5－235513

亞富影音
士丹利街 28 A（陸羽茶樓側）
電話：5－240723

三聯書店讀者服務中心
香港中環域多利皇后街九號（中環街市側）
電話：5~250102

銅鑼灣

百老滙攝影器材有限公司（第十四店）
軒尼斯道 529號地下（三越百貨公司對面）
電話：5－770181

太古城

百老滙攝影器材有限公司（第七店）
太古城中心第 232店
電話：5－672006

西環

創業雜誌中心
西環屈地街22號地下　明愛中心對面
電話：5-478338

旺角

百老滙攝影器材有限公司　（第 1,3,9,10 店）
彌敦道 758號在下（凱聲戲院側）
電話：3－816931
彌敦道700 號旺角中心第二期地下
電話：3－943827
豉油街 26 號地下（近商務書局）
電話：3－884331
亞皆老街 60 B 地下（世界書局對面）
電話：3－956367

深水埗

百老滙攝影器材有限公司（第十三店）
青山道148 號地下（華聲戲院對面）
電話：3－7295938

尖沙咀

智源書局
金巴利道 27－33號永利大厦 2字樓A座
電話：3 － 678482　3－678414

荔枝角

百老滙攝影器材有限公司（第二店）
美孚新邨第四期百老滙街 79號（影都戲院側）
電話：3－7424923

觀塘

百老滙攝影器材有限公司（第四、十二店）
物華街 3 號地下（皇上皇側）
電話：3－419513
同仁街 35 號地下（大家樂側）
電話：3－449940

荃灣

百老滙攝影器材有限公司（第十一店）
沙咀道 230 A（戴麟趾夫人健康院對面）
電話： 0-4938926

屯門

恒達時電腦公司
屯門華都花園地下商場 8 號室
電話：0-252577

新增設特約經銷商：

教育專業人員協會
（佐頓及北角會所）

電腦時代讀者服務部/ CompuHouse

地址：旺角上海街438號同珍商業中心11字樓1104室
（上海街／登打士街交界）　電話：3-312007

補購各期程式磁碟可前往讀者服務部／COMPU HOUSE

Apple is the trademark of Apple Computer Inc.

何必自己動手

免費 由專業人員上門爲你修理微型電腦



請即加入本公司之保養及免費維修服務

24小時熱綫電話服務：3-306839

微型電腦保養維修中心

ROM & RAM computer ltd

SHOP: 582–592, Nathan Road, Basement Shop No. B35 Sino Centre, Kowloon. Tel: 3-306839

OFFICE: Rm. 1905 Witty Coml. Bldg., 1A–1J, Tung Choi St., Kowloon. TEL: 3-848129

Telex: 57551 ROMHX

電腦時代精裝珍藏本

(1-12期合訂本)

四月初隆重出版

隨書附送價值超過150元的購買程式磁碟優待券

精裝珍藏本，顧名思義，是將過去出版的電腦時代第1至12期所有的精彩內容滙集成一巨册，其中包括超過六十個的軟件程式使用說明，數十篇有關各類週邊硬件深入解剖報告、熱門軟件評介及用法文章。是所有擁有APPLE或APPLE兼容電腦的用家必須珍藏的巨獻。

可能你已擁有第一至十二期，但這册精裝珍藏本依然有你值得擁有的價值，因為我們在這册合訂本中加插了多篇從未刊登過的文章及程式。而且我們更隨書附送價值超過150元的購買程式磁碟優待券。

電腦時代精裝珍藏本全書厚度超過200餘頁，採用穩固的膠背釘裝。每册僅售50元。四月初出版。現在開始接受預訂。凡於三月三十一日之前利用下列預訂表格訂閱可獲八折優待。

第一至十二期電腦時代單行本
僅餘小量存貨，歡迎補購。

第一期	全部售罄	第七期	全部售罄
第二期	尚餘小量	第八期	全部售罄
第三期	尚餘小量	第九期	尚餘小量
第四期	全部售罄	第十期	全部售罄
第五期	全部售罄	第11期	全部售罄
第六期	全部售罄	第12期	尚餘小量

電腦時代精裝珍藏本(1-12期合訂本)
預購優待表格

本人欲訂購電腦時代精裝珍藏本　　册，現本人只須付出每册40元的代價。隨函附上支票／銀行本票乙張。

姓名：＿＿＿＿＿＿＿＿

地址：＿＿＿＿＿＿＿＿

＿＿＿＿＿＿＿＿＿＿

電話：＿＿＿＿＿＿＿＿

附上劃線支票No. ＿＿＿＿＿＿＿＿

如要掛號寄出請加付4元。

請用英文正楷書寫，方便電腦處理

學校電腦輔助教育

77 ■ 組合線性方程式解答系統

兒童電腦輔助教育

72 ■ 成雙成對

最新遊戲軟件評介

82 PITSTOP II 油漆佬積集
UP'n DOWN 捉鬼敢死隊
PITFALL II 米奇老鼠星際探險
花生砌圖 TRAPPER醉酒鬼

BACK UP資料館

24 POKE 214, 128之破解方法
35 LIST 程式變BOOT 機
80 ■ 磁碟保護器——COPY VTOC 法
81 DISK MUNCHER妙用與幾個熱門遊戲的抄法

精品店

37 ■ 捉蟲訓練
63 BLOAD位置指標
56 41軌磁碟
87 聰明的RUN
88 ■ 一行程式參加得獎者

讀者園地

75 學校電腦學會活動贊助基金接受申請
89 APPLE FAN CLUB 成立
4 COMPUSOFT磁碟月刊廣告介紹
39 業餘程式編寫員好消息
86 各期程式磁碟總目錄
33 訂閱表格

■ 程式磁碟

特稿

ASCII 大揭秘 15
■ 學習迷宮產生的原理及自編迷宮遊戲 18
BOOT 零程式的活用與分析 84
■ 使用更方便的BOOT DRIVE 2 軟件 23

高級篇

CP/M 版本的VISICALC ——SUPERCALC 36
WORDSTAR 的操作備忘——操作指令總覽表 34

中小型工商業應用

■ 大忙人時間顧問 40

應用軟件庫

■ 專業室內設計師 25

小型製作專欄

16 K RAM 咭改良之(二) ——解決HANG 機的困擾 38

齊齊樂開心坊

■ 太空侵略者封閉戰 64
■ 動態/策略兼備的戰爭遊戲:BEACH HEAD 68
■ 吃不到的葡萄不是酸的!! 70

軟件縱橫談

TAKE 1 同廠產品——PIXIT 應用概論 46
爲II PLUS 而寫的三合一綜合軟件—— JANE 50
BEAGLE BASIC:一種加强版本的BASIC 語言 51

專欄

■ DOS 3.3 專欄—— 兩個新DOS 3.3 指令—MREAD & MWRT(下) 57
機械語言入門—— 機械語言未來的命運 59
■ 麥法基專欄—— 爲你的程式配上立體背景音樂 60

蘋果工具箱

■ LIST 大師 54
■ CATALOG HELPER 52
■ 40 軌磁碟 SECTOR 使用情況顯示 56

電腦時代

NO. 16

本刊每月廿五號出版。一年十二期。全年訂閱費120港元,雜誌連程式磁碟爲475港元。海外訂戶請參看訂閱收費表。
◘出版:電腦時代出版社 ◘督印人:伍小華 ◘編輯:電腦時代編輯部 ◘軟件支援:COMPUHOUSE ◘通訊地址:九龍中央郵箱71193號 ◘通訊電話:3-312007 ◘讀者服務部地址:旺角上海街438號同珍商業中心11字樓1104室(登打士街交界) ◘承印者:永恒印刷公司 ◘發行人:同德書報社



香港代理：華益企業公司
NO.31-33 KING'S ROAD, 1ST FLOOR,
(NEAR LAU LI ST.) NORTH POINT
G.P.O. BOX 10617,
HONG KONG

TELEX NO: 75632 PECO HX
CABLE ADDRESS: "PETCOMY" HONG KONG
TELEPHONE: 5-784928, 5-783129
BANKER: HANG LUNG BANK LTD.